Canonical automata via distributive law homomorphisms

Stefan Zetzsche\textsuperscript{a,2,4} Gerco van Heerdt\textsuperscript{a,3} Matteo Sammartino\textsuperscript{a,b,3} Alexandra Silva\textsuperscript{a,1}

\textsuperscript{a} Department of Computer Science, University College London, London WC1E 6EA, United Kingdom
\textsuperscript{b} Department of Computer Science, Royal Holloway University of London, Egham TW20 0EX, United Kingdom

Abstract

The classical powerset construction is a standard method converting a non-deterministic automaton into a deterministic one recognising the same language. Recently, the powerset construction has been lifted to a more general framework that converts an automaton with side-effects, given by a monad, into a deterministic automaton accepting the same language. The resulting automaton has additional algebraic properties, both in the state space and transition structure, inherited from the monad. In this paper, we study the reverse construction and present a framework in which a deterministic automaton with additional algebraic structure over a given monad can be converted into an equivalent succinct automaton with side-effects. Apart from recovering examples from the literature, such as the canonical residual finite-state automaton and the atomaton, we discover a new canonical automaton for a regular language by relating the free vector space monad over the two element field to the neighbourhood monad. Finally, we show that every regular language satisfying a suitable property parametric in two monads admits a size-minimal succinct acceptor.

1 Introduction

The existence of a unique minimal deterministic acceptor is an important property of regular languages. Establishing a similar result for non-deterministic acceptors is significantly more difficult, but nonetheless of great practical importance, as non-deterministic automata (NFA) can be exponentially more succinct than deterministic ones (DFA). The main issue is that a regular language can be accepted by several size-minimal NFAs that are not isomorphic. A number of sub-classes of non-deterministic automata have been identified in the literature to tackle this issue, which all admit canonical representatives: the atomaton [1], the canonical residual finite-state automaton (short canonical RFSA and also known as jiromaton) [2], the minimal xor automaton [3], and the distromaton [4].

In this paper we provide a general categorical framework that unifies constructions of canonical non-deterministic automata and unveils new ones. Our framework adopts the well-known representation of side-effects via monads [5] to generalise non-determinism in automata. For instance, an NFA (without initial states) can be represented as a pair \( \langle X, k \rangle \), where \( X \) is the set of states and \( k : X \rightarrow 2 \times \mathcal{P}(X)^A \) combines the function classifying each state as accepting or rejecting with the function giving the set of next states for each input. The powerset forms a monad \( \langle \mathcal{P}, \{ - \}, \mu \rangle \), where \( \{ - \} \) creates singleton sets and \( \mu \) takes the union of a set of sets. This allows describing the classical powerset construction, converting an NFA into a DFA, in categorical terms [6] as depicted on the left of Figure 1, where \( k^2 : \mathcal{P}(X) \rightarrow 2 \times \mathcal{P}(X)^A \) represents an equivalent DFA, obtained by taking the subsets of \( X \) as states, and \( (\varepsilon, \delta) : 2^A \rightarrow 2 \times (2^AX)^A \) is the automaton of languages. There then exists a unique automaton homomorphism obs, assigning a language semantics to each set of states.

---

\textsuperscript{1} The author has been supported by the ERC via the Consolidator Grant AutoProbe 101002697 and by a Royal Society Wolfson Fellowship.
\textsuperscript{2} The author has been supported by GCHQ via the VeTSS grant “Automated black-box verification of networking systems” (4207703/RFA 15845).
\textsuperscript{3} The author has been supported by the EPSRC via the Standard Grant CLeVer (EP/S028641/1).
\textsuperscript{4} Email: s.zetzsche@cs.ucl.ac.uk

This paper will be published in the proceedings of MFPS XXXVII

URL: https://www.coalg.org/calco-mfps2021/mfps/
A language is a monad $T$, where $2 \times (-)^A$ is replaced by any (suitable) functor $F$ describing the automaton structure, and $\mathcal{P}$ by a monad $T$ describing the automaton side-effects. $\Omega \xrightarrow{\omega} F\Omega$ is the so-called final coalgebra, providing a semantic universe that generalises the automaton of languages.

2 Overview of the approach

In this section, we give an overview of the ideas of the paper through an example. We show how our methodology allows recovering the construction of the automaton for the regular language $L = (a+b)^*a$, which consists of all words over $A = \{a,b\}$ that end in $a$. For each step, we hint at how it is generalised in our framework.

The classical construction of the automaton for $L$ consists in closing the residuals\(^5\) of $L$ under all Boolean operations, and then forming a non-deterministic automaton whose states are the atoms\(^6\) of the ensuing complete atomic Boolean algebra (CABA)—that is, non-empty intersections of complemented or uncomplemented residuals. In our categorical setting, this construction is obtained in several steps, which we now describe.

2.1 Computing residuals

We first construct the minimal DFA accepting $L$ as a coalgebra of type $M_L \rightarrow 2 \times (M_L)^A$. By the well-known Myhill-Nerode theorem [9], $M_L$ is the set of residuals for $L$. The automaton is depicted in Figure 2.

In our framework, we consider coalgebras over an arbitrary endofunctor $F: \mathcal{C} \rightarrow \mathcal{C}$ ($F = 2 \times (-)^A$ and $\mathcal{C} = \text{Set}$ in this case). Minimal realisations, generalising minimal DFAs, exist for a wide class of functors $F$ and categories $\mathcal{C}$, including all the examples in this paper.

---

\(^5\) A language is a residual or left quotient of $L \subseteq A^*$, if it is of the form $v^{-1}L = \{u \in A^* \mid vu \in L\}$ for some $v \in A^*$.

\(^6\) A non-zero element $a \in B$ is called atom, if for all $x \in B$ such that $x \leq a$ one finds $x = 0$ or $x = a$. 
A Definition 3.1 (Coalgebra)

We close the minimal DFA under all Boolean operations, generating an equivalent deterministic automaton that has additional algebraic structure: its state space is a CABA. This is achieved via a double powerset construction—where sets of sets are interpreted as full disjunctive normal form—and the resulting coalgebra is of type \( P^2(M_L) \to 2 \times (P^2(M_L))^A \). Our construction relies on the so-called "neighbourhood monad" \( \mathcal{H} \), whose algebras are precisely CABAs, and yields a (free) bialgebra capturing both the coalgebraic and the algebraic structure; the interplay of these two structures is captured via a distributive law. We then minimise this DFA to identify Boolean expressions evaluating to the same language. As desired, the resulting state space is precisely the Boolean closure of the residuals of \( \mathcal{L} \). Formally, we obtain the minimal bialgebra for \( \mathcal{L} \) depicted in Figure 3.

This step in our framework is generalised as closure of an \( F \)-coalgebra w.r.t. (the algebraic structured induced by) any monad \( S \) for which a suitable distributive law \( \lambda \) with the coalgebra endofunctor \( F \) exists. The first step of the closure yields a free \( \lambda \)-bialgebra, comprised of both an \( F \)-coalgebra and an \( S \)-algebra over the same state space. In a second step, minimisation is carried out in the category of \( \lambda \)-bialgebras, which guarantees simultaneous preservation of the algebraic structure and of the language semantics.

2.3 Constructing the átomaton

This step is the key technical result of our paper. Atoms have the property that their Boolean closure generates the entire CABA. In our framework, this property is generalised via the notion of generators for algebras over a monad, which allows one to represent a bialgebra as an equivalent free bialgebra over its generators, and hence to obtain succinct canonical representations (Proposition 4.3). In Section 4 we apply this result to obtain the canonical RFSA, the canonical nominal RFSA, and the minimal xor automaton for a given regular language.

However, to recover the átomaton from the minimal CABA-structured DFA of the previous step, in addition a subtle change of perspective is required. In fact, we are still working with the "wrong" side-effect: the non-determinism of bialgebras so far is determined by \( \mathcal{H} \), whereas we are interested in an NFA, whose non-determinism is captured by the powerset monad \( \mathcal{P} \). As is well-known, every element of a CABA can be obtained as the join of the atoms below it. In other words, those atoms are also generators of the underlying CSL, which is an algebra for \( \mathcal{P} \). We formally capture this idea as a map between monads \( \mathcal{H} \to \mathcal{P} \). Crucially, we show that this map lifts to a distributive law homomorphism and allows translating a bialgebra over \( \mathcal{H} \) to a bialgebra over \( \mathcal{P} \), which can be represented as a free bialgebra over atoms—the átomaton for \( \mathcal{L} \), which is shown in Figure 4.

In Section 5 we generalise this idea to the situation of two monads \( S \) and \( T \) involved in distributive laws with the coalgebra endofunctor \( F \). In particular, Corollary 5.3 is our free representation result, spelling out a condition under which a bialgebra over \( S \) can be represented as a free bialgebra over \( T \), and hence admits an equivalent succinct representation as an automaton with side-effects in \( T \). Besides the átomaton and the examples in Section 4, this construction allows us to capture the distromaton and a newly discovered canonical acceptor that relates CABAs with vector spaces over the two element field.

3 Preliminaries

We assume basic knowledge of category theory (including functors, natural transformations, and adjunctions) [10]. In this section we recall the relevant notions for our technical development: coalgebras, monads, algebras over a monad, distributive laws, and bialgebras.

Unpointed deterministic automata are basic examples of coalgebras in the category of sets and functions: they are of the type \( k: X \to FX \), where \( FX = 2 \times X^A \) and \( k \) pairs the final state function and the transition function assigning a next state to each letter \( a \in A \). Coalgebra has emerged as a unifying framework to study infinite data types and state-based systems [11].

Definition 3.1 (Coalgebra) A coalgebra for an endofunctor \( F \) in a category \( \mathcal{C} \) is a tuple \((X,k)\) consisting of an object \( X \) in \( \mathcal{C} \) and a morphism \( k: X \to FX \).

Crucial in the theory of coalgebras is the notion of homomorphism, which allows to relate states of coalgebras of the same type. A homomorphism \( f: (X,k_X) \to (Y,k_Y) \) between \( F \)-coalgebras is a morphism \( f: X \to Y \) satisfying \( k_Y \circ f = Ff \circ k_X \). The category of \( F \)-coalgebras and homomorphisms is denoted by \( \text{Coalg}(F) \). If it exists, the final object of this category is of particular importance.
Definition 3.3 (Monad) A monad on a category \( \mathcal{C} \) is a tuple \( (T, \eta, \mu) \) consisting of an endofunctor \( T : \mathcal{C} \to \mathcal{C} \) and natural transformations \( \eta : \text{id}_{\mathcal{C}} \Rightarrow T \) and \( \mu : T^2 \Rightarrow T \) satisfying \( \mu \circ T \mu = \mu \circ \mu T \) and \( \mu \circ \eta T = T \circ \mu \eta \).

By a slight abuse of notation we will refer to a monad simply by its underlying endofunctor.

Non-determinism is typically modelled by the powerset monad \( \mathcal{P} \), whose underlying endofunctor \( \mathcal{P} \) assigns to a state \( x \) of another form [20]. For our particular case it is sufficient to consider distributive laws between a monad and endofunctor, sometimes referred to as Eilenberg-Moore laws [21].

Example 3.5 • The category \( \text{Alg}(\mathcal{P}) \) is isomorphic to the category of complete join-semi lattices (CSL) and functions that preserve all joins [16].

• The category \( \text{Alg}(\mathcal{H}) \) is isomorphic to the category of complete atomic Boolean algebras (CABA) and Boolean algebra homomorphisms that preserve all meets and all joins [15].

• The category \( \text{Alg}(\mathcal{A}) \) is isomorphic to the category of completely distributive lattices (CDL) and functions that preserve all meets and all joins [15].

• The category \( \text{Alg}(\mathcal{R}) \) is isomorphic to the category of vector spaces over the unique two element field \( \mathbb{Z}_2 \)-Vector) and linear maps [16].

Distributive laws have originally occurred as a way to compose monads [19], but now also exist in a wide range of other forms [20]. For our particular case it is sufficient to consider distributive laws between a monad and an endofunctor, sometimes referred to as Eilenberg-Moore laws [21].
Definition 3.6 (Distributive law) A distributive law between a monad $T$ and an endofunctor $F$ on $\mathcal{C}$ is a natural transformation $\lambda : TF \Rightarrow FT$ satisfying $F\eta_X = \lambda_X \circ \eta_{FX}$ and $\lambda_X \circ \mu_{FX} = F\mu_X \circ \lambda_{TX} \circ T\lambda_X$.

For example, every algebra $h : TB \to B$ for a set monad $T$ induces a distributive law $\lambda^h$ between $T$ and $F$ with $FX = B \times X^A$ defined by
\[
\lambda^h_X := (h \times \text{st}) \circ (T\pi_1, T\pi_2),
\]
where $\text{st}$ denotes the usual strength function. For instance, the algebra structures defined by $\lambda^h_X = \{ (x, y) \mid x \in X, y \in Y \}$, where we identify subsets with their characteristic functions. In these cases we will abuse notation and write $\lambda^F$ instead of $\lambda^{h^F}$.

Example 3.7 (Generalized determinisation [23]) Given a distributive law, one can model the determinisation of a system with dynamics in $T$ by lifting a $T$-coalgebra homomorphism and an endofunctor $F$ between the category of coalgebras with dynamics in $T$ and the category of coalgebras with dynamics in $F$. For instance, if $\lambda : T X, \mu_X \Rightarrow \lambda^F X, F\mu_X$ is induced by the disjunctive $T$-algebra structure that takes the union of languages. The generalized determinisation procedure in Example 3.7 can now be rephrased in terms of a functor $F T$ between the category of coalgebras with dynamics in $T$ and the final $F T$-coalgebra.

The example above illustrates the concept of a bialgebra: the algebraic part $(TX, \mu_X)$ and the coalgebraic part $(TX, k^X)$ of a lifted automaton are compatible along the distributive law $\lambda$.

Definition 3.8 (Bialgebra) A $\lambda$-bialgebra is a tuple $(X, h, k)$ consisting of a $T$-algebra $(X, h)$ and an $F$-coalgebra $(X, k)$ satisfying $Fh \circ \lambda_X \circ Tk = k \circ h$.

A homomorphism between $\lambda$-bialgebras is a morphism between the underlying objects that is simultaneously a $T$-algebra homomorphism and an $F$-coalgebra homomorphism. The category of $\lambda$-bialgebras and homomorphisms is denoted by $\text{Bialg}(\lambda)$. The existence of a final $F$-coalgebra is equivalent to the existence of a final $\lambda$-bialgebra, as the next result shows.

Lemma 3.9 [24] Let $(\Omega, k_\Omega)$ be the final $F$-coalgebra, then $(\Omega, h_\Omega, k_\Omega)$ with $h_\Omega := \text{obs}_{(T\Omega, \lambda_\Omega \circ Tk_\Omega)}$ is the final $\lambda$-bialgebra satisfying $\text{obs}_{(X, h, k)} = \text{obs}_{(X, k)}$. Conversely, if $(\Omega, h_\Omega, k_\Omega)$ is the final $\lambda$-bialgebra, then $(\Omega, k_\Omega)$ is the final $F$-coalgebra.

For instance, for the distributive law in Example 3.7, the final bialgebra is carried by the final coalgebra $\mathcal{P}(A^*)$ and also has a free $\mathcal{P}$-algebra structure that takes the union of languages.

The generalized determinisation procedure in Example 3.7 can now be rephrased in terms of a functor between the category of coalgebras with dynamics in $F$ and side-effects in $T$ on the one side, and the category of bialgebras on the other side.

Lemma 3.10 [24] Defining $\text{exp}_T((X, k)) := (TX, \mu_X, (F\mu_X \circ \lambda_{TX}) \circ Tk)$ and $\text{exp}_T(f) := Tf$ yields a functor $\text{exp}_T : \text{Coalg}(FT) \to \text{Bialg}(\lambda)$.

We will sometimes refer to the functor which arises from the one above by precomposition with the canonical embedding of $F$-coalgebras into $FT$-coalgebras.

Corollary 3.11 Defining $\text{free}_T((X, k)) := (TX, \mu_X, \lambda_X \circ Tk)$ and $\text{free}_T(f) := Tf$ yields a functor $\text{free}_T : \text{Coalg}(F) \to \text{Bialg}(\lambda)$ satisfying $\text{free}_T((X, k)) = \text{exp}_T((X, F\mu_X \circ k))$.

4 Succinct automata from bialgebras

In this section we introduce the foundations of our theoretical contributions. We begin with the notion of a generator [8] for an algebra over a monad and demonstrate how it can be used to translate a bialgebra into an equivalent free bialgebra. While the treatment is very general, we are particularly interested in the case in

---

8 For any two sets $X, A$ the strength function $\text{st} : T(X^A) \to (TX)^A$ is defined by $\text{st}(U)(a) = T(\text{ev}_a)(U)$, where $\text{ev}_a(f) = f(a)$.
A non-zero element $x \in L$ is called join-irreducible if for all $y, z \in L$ such that $x = y \lor z$ one finds $x = y$ or $x = z$.

Minimal in the sense that every state is reachable by an element of $A^*$ and no two different states observe the same language.

\[ A \cdot b \quad b \quad a \quad (\lor) \quad h \quad (\lor) \quad a \quad (\lor) \quad h \quad (\lor) \quad a \quad (\lor) \quad a, b \]

Figure 5. (a) The minimal CSL-structured DFA for $L = (a + b)^* a$; (b) The canonical RFSA for $L = (a + b)^* a$.

which the bialgebra is given by a deterministic automaton that has additional algebraic structure over a given monad, and the translation results in an automaton with side-effects in that monad. We will demonstrate that the theory in this section instantiates to the canonical RFSA [2], the canonical nominal RFSA [14], and the minimal xor automaton [3].

Definition 4.1 (Generator and basis) A generator for a $T$-algebra $(X, h)$ is a tuple $(Y, i, d)$ consisting of an object $Y$, a morphism $i : Y \to X$, and a morphism $d : X \to TY$ such that $(h \circ Ti) \circ d = \text{id}_X$. A generator is called a basis if it additionally satisfies $d \circ (h \circ Ti) = \text{id}_Y$.

A generator for an algebra is called a scoop by Arbib and Manes [8]. Here, we additionally introduce the notion of a basis. Intuitively, one calls a set $Y$ that is embedded into an algebraic structure $X$ a generator for the latter if every element $x \in X$ admits a decomposition $d(x) \in TY$ into a formal combination of elements of $Y$ that evaluates to $x$. If the decomposition is moreover unique, that is, $h \circ Ti$ is not only a surjection with right-inverse $d$, but a bijection with two-sided inverse $d$, then a generator is called a basis. Every algebra is generated by itself using the generator $(X, \text{id}_X, \eta_Y)$, but not every algebra admits a basis. We are particularly interested in classes of set-based algebras for which every algebra admits a size-minimal generator, that is, no generator has a carrier of smaller size. In such a situation we will also speak of canonical generators.

Example 4.2 • A tuple $(Y, i, d)$ is a generator for a $P$-algebra $L = \langle X, h, \cdot \rangle \simeq \langle X, \lor^h \rangle$ iff $x = \bigvee_{y \in d(x)} i(y)$ for all $x \in X$. Note that if $Y \subseteq X$ is a subset, then $i(y) = y$ for all $y \in Y$. If $L$ satisfies the descending chain condition, which is in particular the case if $X$ is finite, then defining $i(a) = a$ and $d(x) = \{ y \in J(L) \mid y \leq x \}$ turns the set of join-irreducibles $J(L)$ into a size-minimal generator $(J(L), i, d)$ for $L$, cf. Lemma B.15.

• A tuple $(Y, i, d)$ is a generator for a $\mathcal{R}$-algebra $V = \langle X, h \rangle \simeq \langle X, +^h, h \rangle$ iff $x = \sum_{y \in Y} d(x)(y)^h i(y)$ for all $x \in X$. As it is well-known that every vector space can be equipped with a basis, every $\mathcal{R}$-algebra $V$ admits a basis. One can show that a basis is size-minimal, cf. Lemma B.12.

It is enough to find generators for the underlying algebra of a bialgebra to derive an equivalent free bialgebra. This is because the algebraic and coalgebraic components are tightly intertwined via a distributive law.

Proposition 4.3 Let $(X, h, k)$ be a $\lambda$-bialgebra and let $(Y, i, d)$ be a generator for the $T$-algebra $(X, h)$. Then $h \circ Ti : \exp T((Y, Fd \circ k \circ i)) \to (X, h, k)$ is a $\lambda$-bialgebra homomorphism.

Intuitively, the bialgebra $(X, h, k)$ is a deterministic automaton with additional algebraic structure in the monad $T$ and say initial state $x \in X$, while the equivalent free bialgebra is the determinisation of the succinct automaton $Fd \circ k \circ i : Y \to FTY$ with side-effects in $T$ and initial state $d(x) \in TY$. The following result further observes that if one considers a basis for the underlying algebraic structure of a bialgebra, rather than just a generator, then the equivalent free bialgebra is in fact isomorphic to the original bialgebra.

Proposition 4.4 Let $(X, h, k)$ be a $\lambda$-bialgebra and let $(Y, i, d)$ be a basis for the $T$-algebra $(X, h)$. Then $h \circ Ti : \exp T((Y, Fd \circ k \circ i)) \to (X, h, k)$ is a $\lambda$-bialgebra isomorphism.

We conclude this section by illustrating how Proposition 4.3 can be used to construct the canonical RFSA [2], the canonical nominal RFSA [14], and the minimal xor automaton [3] for a regular language $L$ over some alphabet $A$. All examples follow three analogous steps:

(i) We construct the minimal pointed coalgebra $M_L$ for the (nominal) set endofunctor $F = 2 \times (-)^4$ accepting $L$. For the case $A = \{ a, b \}$ and $L = (a + b)^* a$, the coalgebra $M_L$ is depicted in Figure 2.

(ii) We equip the former with additional algebraic structure in a monad $T$ (which is related to $F$ via a canonically induced distributive law $\lambda$) by generating the $\lambda$-bialgebra free$_T(M_L)$. By identifying semantically
The Example 4.7 (The minimal xor automaton) induced by Proposition 4.3 is depicted in Figure 6.

The underlying vector space structure necessarily has a basis; we choose the size-minimal generator consisting of
\[ v, w, u \in A^*, a \in A \].

We identify canonical generators \( \langle Y, i, d \rangle \) for \( (X, h) \) and use Proposition 4.3 to derive an equivalent succinct automaton \( \langle Y, Fd \circ k \circ i \rangle \) with side-effects in \( T \).

Example 4.5 (The canonical RFSA) Using the \( P \)-algebra structure \( h^P : \mathcal{P}2 \to 2 \) with \( h^P(\varphi) = \varphi(1) \), we derive a canonical distributive law \( \lambda^P \) between \( F \) and the powerset monad \( \mathcal{P} \). The minimal pointed \( \lambda^P \)-bialgebra for \( \mathcal{L} = (a + b)^*a \) with its underlying CSL structure is depicted in Figure 5a; the construction can be verified with the help of Lemma B.7. The partially ordered state space \( L = \{ [\emptyset] \leq [(x)] \leq [(y)] \} \) is necessarily finite, thus satisfies the descending chain condition, which turns the set of join-irreducibles into a size-minimal generator \( \langle J(L), i, d \rangle \) with \( i(y) = y \) and \( d(x) = \{ y \in J(L) \mid y \leq x \} \), cf. Lemma B.15. In this case, the join-irreducibles are given by all non-zero states. The \( \mathcal{P} \)-succinct automaton consequently induced by Proposition 4.3 is depicted in Figure 5b; it can be recognised as the canonical RFSA, cf. e.g. [4].

Example 4.6 (The canonical RFSA) It is not hard to see that \( F \) extends to a functor on the category of nominal sets; the usual strength function is equivariant (Lemma B.6); and \( h^{P_n} : \mathcal{P}_n2 \to 2 \) with \( h^{P_n}(\varphi) = \varphi(1) \) defines a \( P_n \)-algebra, which induces a canonical distributive law \( \lambda^{P_n} \) between \( F \) and the nominal powerset monad \( \mathcal{P}_n \). As in [14], let \( \mathcal{L} = \{ vauw \mid v, w, u \in A^*, a \in A \} \), then \( a^{-n}\mathcal{L} = a^{-2}\mathcal{L} = A^* \) for \( n \geq 2 \), and \( v^{-1}\mathcal{L} = \bigcup_{a \in A} a^{-1}v(a)\mathcal{L} \), where \( |v(a)| \) denotes the number of \( a \)'s that occur in \( v \). In consequence, the nominal CSL underlying the minimal pointed \( \lambda^{P_n} \)-bialgebra is generated by the orbit-finite nominal set of join-irreducibles \( \langle \mathcal{L} \rangle \cup \{ a^{-1}\mathcal{L} \mid a \in A \} \cup \{ A^* \} \), which is equipped with the obvious \( \text{Perm}(A) \)-action and satisfies the inclusion \( \mathcal{L} \subseteq a^{-1}\mathcal{L} \subseteq A^* \). The orbit-finite representation of the \( P_n \)-succinct automaton consequently induced by Proposition 4.3 is depicted in Figure 6.

Example 4.7 (The minimal xor automaton) The \( R \)-algebra structure \( h^R : \mathcal{R}2 \to 2 \) with \( h^R(\varphi) = \varphi(1) \) induces a canonical distributive law \( \lambda^R \) between \( F \) and the free vector space monad \( \mathcal{R} \) over the two element field. The minimal pointed \( \lambda^R \)-bialgebra accepting \( \mathcal{L} = (a + b)^*a \) is depicted in Figure 7a and coincides with the bialgebra freely generated by the \( F \)-coalgebra in Figure 2. The construction can be verified using Lemma B.10. The underlying vector space structure necessarily has a basis; we choose the size-minimal generator consisting of \( \{ x \} \) and \( \{ x, y \} \), cf. Lemma B.12. The well-definedness of our choice is witnessed by \( d(\emptyset) := \emptyset, d(\{ x \}) := \{ \{ x \} \} \), \( d(\{ y \}) := \{ \{ x \}, \{ x, y \} \} \), and \( d(\{ x, y \}) := \{ \{ x, y \} \} \). The \( R \)-succinct automaton induced by Proposition 4.3 is depicted in Figure 7b; it can be recognised as the minimal xor automaton, cf. e.g. [4].

5 Changing the type of succinct automata

This section contains a generalisation of the approach in Section 4. The extension is based on the observation that in the last section we implicitly considered two types of monads: (i) a monad \( S \) that describes the additional algebraic structure of a given deterministic automaton; and (ii) a monad \( T \) that captures the side-effects of the succinct automaton that is obtained by the generator-based translation. In Proposition 4.3, the main result of the last section, the monads coincided, but to recover for instance the áutomaton [1] we will have to extend Proposition 4.3 to a situation where \( S \) and \( T \) can differ.

5.1 Relating distributive laws

We now introduce the main technical ingredient of our extension: distributive law homomorphisms. As before, we present the theory on the level of arbitrary bialgebras, even though we will later focus on the case where the coalgebraic dynamics are those of deterministic automata. Distributive law homomorphisms will allow us to shift a bialgebra over a monad \( S \) to an equivalent bialgebra over a monad \( T \), for which we can then find, analogous to Section 4, an equivalent succinct representation. The notion we use is an instance of a much more general definition that allows to relate distributive laws on two different categories. We restrict to the case where both distributive laws are given over the same behavioural endofunctor \( F \).

11 Minimal in the sense that every state is reachable by an element of \( T(A^*) \) and no two different states observe the same language.
Corollary 5.3 Let \( \lambda^S : SF \to FS \) and \( \lambda^T : TF \to FT \) be distributive laws between monads \( S \) and \( T \) and an endofunctor \( F \), respectively. A distributive law homomorphism \( \alpha : \lambda^S \to \lambda^T \) consists of a natural transformation \( \alpha : T \Rightarrow S \) satisfying \( \mu^S \circ \alpha_S \circ T \alpha = \alpha \circ \mu^T \), \( \alpha \circ \eta^T = \eta^S \), and \( \lambda^S \circ \alpha_F = F \alpha \circ \lambda^T \).

The above definition is such that \( \alpha \) induces a functor between the categories of \( \lambda^S \)- and \( \lambda^T \)-bialgebras.

Lemma 5.2 [27, 28] Let \( \alpha : \lambda^S \to \lambda^T \) be a distributive law homomorphism. Then \( \alpha(\langle X, h, k \rangle) := \langle X, h \circ \alpha_X, k \rangle \) and \( \alpha(f) := f \) defines a functor \( \alpha : \text{Bialg}(\lambda^S) \to \text{Bialg}(\lambda^T) \).

The next result is a straightforward consequence of Proposition 4.3, and may be strengthened to an isomorphism in case one is given a basis instead of a generator, analogous to Proposition 4.4. It can be seen as a road map to the approach we propose in this section.

Corollary 5.3 Let \( \alpha : \lambda^S \to \lambda^T \) be a homomorphism between distributive laws and \( \langle X, h, k \rangle \) a \( \lambda^S \)-bialgebra. If \( (Y, i, d) \) is a generator for the \( T \)-algebra \( \langle X, h \circ \alpha_X \rangle \), then \( (h \circ \alpha_X) \circ T i : \text{exp}_T((Y, P d \circ k \circ i)) \to \langle X, h \circ \alpha_X \rangle \) is a \( \lambda^T \)-bialgebra homomorphism.

5.2 Deriving distributive law relations

We now turn to the procedure of deriving a distributive law homomorphism. In practice, coming up with a natural transformation and proving that it lifts to a distributive law homomorphism can be quite cumbersome.

Fortunately, for certain cases, there is a way to simplify things significantly. For instance, as the next result shows, if, as in (1), the involved distributive laws are induced by algebra structures \( h^S \) and \( h^T \) for an output set \( B \), respectively, then one of the conditions is implied by a less convoluted constraint.

Lemma 5.4 Let \( \alpha : T \Rightarrow S \) be a natural transformation satisfying \( h^S \circ \alpha_B = h^T \), then \( \lambda^S \circ \alpha_F = F \alpha \circ \lambda^T \).

The next result shows that for the neighbourhood monad there exists a family of canonical choices of distributive law homomorphisms parametrised by Eilenberg-Moore algebra structures on the output set \( B = 2 \). While it is well-known that such algebras induce a monad morphism, for instance in the coalgebraic modal logic community [29–31], its commutativity with canonical distributive laws has not been observed before. Moreover, we provide a new formalisation in terms of the strength function, which allows the result to be lifted to strong monads and arbitrary output objects on other categories than the one of sets and functions.

Proposition 5.5 Any algebra \( h : T2 \to 2 \) over a set monad \( T \) induces a homomorphism \( \alpha^h : \lambda^H \to \lambda^B \) between distributive laws by \( \alpha^h_X := h^2 \circ \text{st} \circ T(\eta^H_X) \).

The rest of the section is concerned with using Proposition 5.5 and Corollary 5.3 to derive canonical acceptors based on induced distributive law homomorphisms.

5.3 Example: The atomic

We will now justify the previous informal construction of the atomic. As hinted before, the atomic can be recovered by relating the neighbourhood monad \( \mathcal{H} \)—whose algebras are complete atomic Boolean algebras (CABAs)—and the powerset monad \( \mathcal{P} \). Formally, as a consequence of Proposition 5.5 we obtain the following.

Corollary 5.6 Let \( \alpha_X : \mathcal{P}X \to \mathcal{H}X \) satisfy \( \alpha_X(\varphi)(\psi) = \bigvee_{x \in X} \varphi(x) \land \psi(x) \), then \( \alpha_X \) constitutes a distributive law homomorphism \( \alpha : \lambda^H \to \lambda^P \).
The next statement follows from a well-known Stone-type duality [32] representation theorem for CABAs.

**Lemma 5.7** Let \( \alpha_X : \mathcal{P}X \to \mathcal{H}X \) satisfy \( \alpha_X(\varphi)(\psi) = \bigvee_{x \in X} \varphi(x) \land \psi(x) \). If \( B = \langle X, h \rangle \) is a \( \mathcal{H} \)-algebra, then \( \langle \text{At}(B), i, d \rangle \) with \( i(a) = a \) and \( d(x) = \{ a \in \text{At}(B) \mid a \leq x \} \) is a basis for the \( \mathcal{P} \)-algebra \( \langle X, h \circ \alpha_X \rangle \).

The â†œmaton for the regular language \( L = (a + b)^*a \), for example, can now be obtained as follows. First, we construct the minimal pointed \( \lambda^A \)-bialgebra accepting \( L \), which is depicted in Figure 3 together with its underlying CABSA structure \( B \). The construction can be verified with the help of Lemma 5.8. Using the distributive law homomorphism \( \alpha \) of Corollary 5.6, it can be translated into an equivalent pointed \( \lambda^B \)-bialgebra with underlying CSL-structure \( \alpha(B) \). By Lemma 5.7 the atoms \( \text{At}(B) \) of \( B \) form a basis for \( \alpha(B) \). In this case the atoms are given by \( \{(x, \{x, y\})\}, \{(y, \emptyset)\} \). The \( \mathcal{P} \)-succinct automaton consequently induced by Corollary 5.3 is depicted in Figure 4; it can be recognised as the â†œmaton, cf. e.g. [4].

### 5.4 Example: The distribaton

We shall now use our framework to recover another canonical non-deterministic acceptor: the distribaton [4]. As the name suggests, it can be constructed by relating the monotone neighbourhood monad \( A \)—whose algebras are completely distributive lattices—and the powerset monad \( \mathcal{P} \). Formally, the relationship can be established by the same natural transformation we used for the â†œmaton.

**Corollary 5.8** Let \( \alpha_X : \mathcal{P}X \to \mathcal{A}X \) satisfy \( \alpha_X(\varphi)(\psi) = \bigvee_{x \in X} \varphi(x) \land \psi(x) \), then \( \alpha \) constitutes a distributive law homomorphism \( \alpha : \lambda^A \to \lambda^P \).

The distribaton for the regular language \( L = (a + b)^*a \), for example, can now be obtained as follows. First, we construct the minimal pointed \( \lambda^A \)-bialgebra for \( L \), depicted in Figure 8a with its underlying CDL structure \( h \). The construction can be verified with the help of Lemma 5.9. Using the distributive law homomorphism \( \alpha \) in Corollary 5.8, it can be translated into an equivalent pointed \( \lambda^B \)-bialgebra with underlying CSL-structure \( \alpha(L) \). Its partially ordered state space \( \{0\} \leq \{(x, \emptyset)\} \leq \{(x, \{x, y\})\} \leq \{(\{x\}, \{x, y\})\} \leq \{\emptyset\} \) is necessarily finite, which turns the set of join-irreducibles into a size-minimal generator \( \langle J(L), i, d \rangle \) for \( L \), where \( i(y) = y \) and \( d(x) = \{ y \in J(L) \mid y \leq x \} \). In this case, the join-irreducibles are given by all non-zero states. The \( \mathcal{P} \)-succinct automaton consequently induced by Corollary 5.3 is depicted in Figure 8b and can be recognised as the distribaton, cf. [4].

### 5.5 Example: The minimal xor-CABA automaton

We conclude this section by relating the neighbourhood monad \( \mathcal{H} \) with the free vector space monad \( \mathcal{R} \) over the unique two element field \( \mathbb{Z}_2 \). In particular, we derive a new canonical succinct acceptor for regular languages, which we call the minimal xor-CABA automaton.

Intuitively, the next result says that every CABA can be equipped with a symmetric difference like operation that turns it into a vector space over the two element field.

**Corollary 5.9** Let \( \alpha_X : \mathcal{R}X \to \mathcal{H}X \) satisfy \( \alpha_X(\varphi)(\psi) = \bigoplus_{x \in X} \varphi(x) \cdot \psi(x) \), then \( \alpha \) constitutes a distributive law homomorphism \( \alpha : \lambda^H \to \lambda^R \).

---

**Figure 8.** (a) The minimal CDL-structured DFA for \( L = (a + b)^*a \), where \( 1 \equiv \{(x, \{x, y\})\}, 2 \equiv \emptyset, 3 \equiv \{(x, \{x, y\})\}, 4 \equiv \{(x, \{y\}, \emptyset)\} \); (b) The distromaton for \( L = (a + b)^*a \).
Since every vector space admits a basis, above result leads to the definition of a new acceptor of regular languages. Let $\alpha$ denote the homomorphism in Corollary 5.9 and $F$ the endofunctor given by $FX = 2 \times X^A$.

**Definition 5.10 (Minimal xor-CABA automaton)** Let $(X, h, k)$ be the minimal $x$-pointed $\lambda^X$-bialgebra accepting a regular language $L \subseteq A^*$, and $B = \{Y, i, d\}$ a basis for the $R$-algebra $(X, h \circ \alpha_X)$. The minimal xor-CABA automaton for $L$ with respect to $B$ is the $d(x)$-pointed $\mathbb{Z}_2$-weighted automaton $Fd \circ k \circ i$.

In Figure 9 it is indicated how the canonical acceptors of this paper, including the minimal xor-CABA automaton, are based on relations between pairs of monads.

For the regular language $L = (a + b)^*$ above definition instantiates as follows. First, as for the automaton, we construct the minimal pointed $\lambda^X$-bialgebra $(X, h, k)$ for $L$; it is depicted in Figure 3. As one easily verifies, the $\mathbb{Z}_2$-vector space $\langle X, h \circ \alpha_X \rangle$ is induced by the symmetric difference operation $\oplus$ on subsets. Using the notation in Figure 3, we choose the basis $\langle Y, i, d \rangle$ with $Y = \{4, 6, 7, 8\}$; $i(y) = y$; and $d(1) = 7 \oplus 8$, $d(2) = \emptyset$, $d(3) = 6 \oplus 7$, $d(4) = 4$, $d(5) = 6 \oplus 7 \oplus 8$, $d(6) = 6$, $d(7) = 7$, $d(8) = 8$. The induced $d(1) = 7 \oplus 8$-pointed $R$-sufficient automaton accepting $L$, i.e. the minimal xor-CABA automaton, is depicted in Figure 10.

### 6 Minimality

In this section we restrict ourselves to the category of (nominal) sets. We show that every language satisfying a certain property is accepting a minimal pointed $\lambda^X$-bialgebra $M$ for $L$ and the underlying algebra of $\alpha(M)$ admits a size-minimal generator, there exists a pointed $\alpha$-closed $T$-sufficient automaton $X$ accepting $L$ such that:

- for any pointed $\alpha$-closed $T$-sufficient automaton $Y$ accepting $L$ we have that $\text{im}(\text{obs}_{\exp_T}(X)) \subseteq \text{im}(\text{obs}_{\exp_T}(Y))$;
- if $\text{im}(\text{obs}_{\exp_T}(X)) = \text{im}(\text{obs}_{\exp_T}(Y))$, then $|X| \leq |Y|$, where $X$ and $Y$ are the carriers of $X$ and $Y$, respectively.

**Figure 9.** The minimal xor-CABA automaton is to the minimal xor automaton what the atomaton is to the canonical RFSA.
Corollary 6.7 The minimal xor automaton for $\mathcal{L} = (a + b)^*a$.  

For a $T$-succinct automaton $\mathcal{X}$ let us write $\text{obs}^\dagger_{\mathcal{X}} := \text{obs}^{\exp_T(\mathcal{X})} \circ \eta^T_X : X \to \Omega$ for a generalisation of the semantics of non-deterministic automata. The next result provides an equivalent characterisation of $\alpha$-closedness in terms of $\text{obs}^\dagger$ that will be particularly useful in Section 6.1.

Lemma 6.5 Let $\alpha : \lambda^S \to \lambda^T$ be a distributive law homomorphism and $(\Omega, h, k)$ the final $\lambda^S$-bialgebra. For any $T$-succinct automaton $\mathcal{X}$ it holds that $\text{im}(\text{obs}_{\exp_T(\mathcal{X})}) = \text{im}(h \circ \alpha_T \circ T(\text{obs}^\dagger_{\mathcal{X}}))$ and $\text{im}(\text{obs}_{\alpha(\text{ext}(\mathcal{X}))}) = \text{im}(h \circ S(\text{obs}^\dagger_{\mathcal{X}}))$.

6.1 Applications to canonical automata

In this section we instantiate Theorem 6.4 to characterise a variety of canonical acceptors from the literature as size-minimal representatives among subclasses of $\alpha$-closed succinct automata, i.e. those automata whose images of the two semantics induced by $\alpha$ coincide. We begin with the canonical RFSA and the minimal xor automaton, for which $\alpha$ is the identity and $\alpha$-closedness therefore is trivial.

In [2] the canonical RFSA for $\mathcal{L}$ has been characterised as size-minimal among those NFAs accepting $\mathcal{L}$ for which states accept a residual of $\mathcal{L}$. More recently, it was shown that the class in fact can be extended to those NFAs accepting $\mathcal{L}$ for which states accept a union of residuals of $\mathcal{L}$ [4]. The next result recovers the latter as a consequence of the second point in Theorem 6.4. We write $\mathcal{Y}$ for the algebraic closure $^12$ of a subset $Y \subseteq X$ of some $T$-algebra $X$.

Corollary 6.6 The canonical RFSA for $\mathcal{L}$ is size-minimal among non-deterministic automata $\mathcal{Y}$ accepting $\mathcal{L}$ with $\text{im}(\text{obs}^\dagger_{\mathcal{Y}})_{\text{CSL}} \subseteq \text{Der}(\mathcal{L})_{\text{CSL}}$.

The second condition in Theorem 6.4 is always satisfied for a reachable succinct automaton $\mathcal{Y}$. Since for $\mathbb{Z}_2$-weighted automata it is possible to find an equivalent reachable $\mathbb{Z}_2$-weighted automaton with less or equally many states (which for NFA is not necessarily the case), the minimal xor automaton is minimal among all $\mathbb{Z}_2$-weighted automata, as was already known from for instance [3].

Corollary 6.7 The minimal xor automaton for $\mathcal{L}$ is size-minimal among $\mathbb{Z}_2$-weighted automata accepting $\mathcal{L}$.

For the automaton, the distromaton, and the minimal xor-CABA automaton the distributive law homomorphism $\alpha$ in play is non-trivial; $\alpha$-closedness translates to the below equalities between closures. In all three cases it is possible to waive the inclusion induced by the second point in Theorem 6.4.

Corollary 6.8 The automaton for $\mathcal{L}$ is size-minimal among non-deterministic automata $\mathcal{Y}$ accepting $\mathcal{L}$ with $\text{im}(\text{obs}^\dagger_{\mathcal{Y}})_{\text{CABA}} = \text{im}(\text{obs}^\dagger_{\mathcal{Y}})_{\text{CSL}}$.

The above result can be shown to be similar to [4, Theorem 4.9], which characterises the automaton as size-minimal among non-deterministic automata whose accepted languages are closed under complement. The result below is very similar to a characterisation of the distromaton as size-minimal among non-deterministic automata whose accepted languages are closed under intersection [4, Theorem 4.13].

Corollary 6.9 The distromaton for $\mathcal{L}$ is size-minimal among non-deterministic automata $\mathcal{Y}$ accepting $\mathcal{L}$ with $\text{im}(\text{obs}^\dagger_{\mathcal{Y}})_{\text{CDL}} = \text{im}(\text{obs}^\dagger_{\mathcal{Y}})_{\text{CSL}}$.

The size-minimality result for the newly discovered minimal xor-CABA automaton is analogous to the ones for the automaton and the distromaton.

Corollary 6.10 The minimal xor-CABA automaton for $\mathcal{L}$ is size-minimal among $\mathbb{Z}_2$-weighted automata $\mathcal{Y}$ accepting $\mathcal{L}$ with $\text{im}(\text{obs}^\dagger_{\mathcal{Y}})_{\text{Z}_2\text{-Vect}} = \text{im}(\text{obs}^\dagger_{\mathcal{Y}})_{\text{CABA}}$.

$^12$ If $Y = \text{im}(f)$ for some morphism $f$ with codomain $(X, h)$, the closure is given by the induced $T$-algebra structure on $\text{im}(h \circ T f)$. 

Figure 10. The minimal xor-CABA automaton for $\mathcal{L} = (a + b)^*a$. 

ZETZSCHE ET AL.
We conclude with a size-comparison between acceptors that is parametric in the closure of derivatives.

**Corollary 6.11** • If \( \text{Der}(L)^{\text{Z}_2, \text{Vect}} = \text{Der}(L)^{\text{CABA}} \), then the minimal xor automaton and the minimal xor-CABA automaton for \( L \) are of the same size.

• If \( \text{Der}(L)^{\text{CSDL}} = \text{Der}(L)^{\text{CDL}} \), then the canonical RFSA and the distromaton for \( L \) are of the same size.

• If \( \text{Der}(L)^{\text{CSDL}} = \text{Der}(L)^{\text{CABA}} \), then the canonical RFSA and the \( \acute{\text{a}} \text{tomaton} \) for \( L \) are of the same size.

7 Related work

One of the main motivations for the present paper is provided by active learning algorithms for the derivation of succinct state-based models [33]. A major challenge in learning non-deterministic models is the lack of a canonical target acceptor for a given language [2]. The problem has been independently approached for different variants of non-determinism, often with the idea of finding a subclass admitting a unique representative [34,35] such as e.g. the canonical RFSA, the minimal xor automaton, or the \( \acute{\text{a}} \text{tomaton} \).

A more general and unifying perspective on learning automata that may not have a canonical target was given by Van Heerdt [36–38]. One of the central notions in this work is the concept of a scoop, originally introduced by Arbib and Manes [8] and here referred to as a generator. The main contribution in [36] is a general procedure to find irreducible sets of generators, which thus restricts the work to the category of sets. In the present paper we generally work over arbitrary categories, although we assume the existence of a minimal set-based generator in Theorem 6.4. Furthermore, the work of Van Heerdt has no size-minimality results.

Closely related to the present paper is the work of Myers et al. [4], who present a coalgebraic construction for canonical non-deterministic automata. They cover the canonical RFSA, the minimal xor automaton, the \( \acute{\text{a}} \text{tomaton} \), and the distromaton. The underlying idea in [4] for finding succinct representations is similar to ours: first they build the minimal DFA for a regular language in a locally finite variety, then they apply an equivalence between the category of finite algebras and a suitable category of finite structured sets and relations. On the one hand, the category of finite algebras in a locally finite variety can be translated into our setting by considering a category of algebras over a monad preserving finite sets. In fact, modulo this translation, many of the categories considered here already appear in [4], e.g. vector spaces, Boolean algebras, complete join-semi lattices, and distributive lattices. On the other hand, their construction seems to be restricted to the category of sets and non-deterministic automata, while we work over arbitrary monads on arbitrary categories. Their work does not provide a general algorithm to construct a succinct automaton, i.e., the specifics vary with the equivalences considered, while we give a general definition and a soundness argument in Corollary 5.3. While Myers et al. give minimality results for a wide range of acceptors, each proof follows case-specific arguments. In Theorem 6.4 we provide a unifying minimality result for succinct automata that generalises parts of [4, Theorem 4.8] and subsumes most of their results [4, Theorem 4.9, Theorem 4.10, Corollary 4.11, Theorem 4.13].

8 Discussion and future work

We have presented a general categorical framework based on bialgebras and distributive law homomorphisms for the derivation of canonical automata. The framework instantiates to a wide range of well-known examples from the literature and allowed us to discover a previously unknown canonical acceptor for regular languages. Finally, we presented a theorem that subsumes previously independently proven minimality results for canonical acceptors, implied new characterisations, and allowed us to make size-comparisons between canonical automata.

In the future, we would like to cover other examples, such as the canonical probabilistic RFSA [34] and the canonical alternating RFSA [35, 39]. Probabilistic automata of the type in [34] are typically modelled as TF-coalgebras instead of PT-coalgebras [24], and thus will need a shift in perspective. For alternating RFSA we expect a canonical form can be constructed in the spirit of this paper, from generators for algebras over the neighbourhood monad, by interpreting the join-dense atoms of a CABA as a full meet of ground elements.

Generally, it would be valuable to have a more systematic treatment of the range of available monads and distributive law homomorphisms [8], making use of the fact that distributive law homomorphisms compose.

Further generalisation in another direction could be achieved by distributive laws between monads and endofunctors on different categories. For instance, we expect that operations on automata as the product can be captured by homomorphisms between distributive laws of such more general type.

Finally, we would like to lift existing double-reversal characterisations of the minimal DFA [41], the \( \acute{\text{a}} \text{tomaton} \) [1], the distromaton [4], and the minimal xor automaton [3] to general canonical automata. The work in [42,43] gives a coalgebraic generalisation of Brzozowski’s algorithm based on dualities between categories, but does not cover the cases we are interested in. The framework in [44] recovers the \( \acute{\text{a}} \text{tomaton} \) as the result of a minimisation procedure, but does not consider other canonical acceptors.
References


A Definitions

Definition A.1 • The powerset monad \( \langle \mathcal{P}, \eta^\mathcal{P}, \mu^\mathcal{P} \rangle \) on the category of sets is given by \( \mathcal{P}X = 2^X \), \( \mathcal{P}f(\varphi)(y) = \bigvee_{x \in \mathcal{P} \mathcal{P}^{-1}(y)} \varphi(x) \), \( \eta^\mathcal{P}_X(x)(y) = [x = y] \), and \( \mu^\mathcal{P}_X(\Phi)(x) = \bigvee_{\varphi \in 2^X} \Phi(\varphi) \land \varphi(x) \).

• The neighbourhood monad \( \langle \mathcal{H}, \eta^\mathcal{H}, \mu^\mathcal{H} \rangle \) on the category of sets is given by \( \mathcal{H}X = 2^{2^X} \), \( \mathcal{H}f(\Phi)(\varphi) = \Phi(\varphi \circ f) \), \( \eta^\mathcal{H}_X(x)(\varphi) = \varphi(x) \), and \( \mu^\mathcal{H}_X(\Psi)(\varphi) = \Psi(\eta^\mathcal{H}_X(\varphi)) \).

• The monotone neighbourhood monad \( \langle \mathcal{A}, \eta^\mathcal{A}, \mu^\mathcal{A} \rangle \) on the category of sets is given by \( \mathcal{A}X = \langle 2, \leq \rangle^{2^{2^X}} \) and otherwise coincides with the neighbourhood monad.

• The free vector space monad over the unique two element field \( \langle \mathcal{R}, \eta^\mathcal{R}, \mu^\mathcal{R} \rangle \) on the category of sets is given by \( \mathcal{R}X = 2^X \), \( \mathcal{R}f(\varphi)(y) = \bigoplus_{x \in \mathcal{P} \mathcal{P}^{-1}(y)} \varphi(x) \), where \( a \oplus b := a + b \mod 2 \), \( \eta^\mathcal{R}_X(x)(y) = [x = y] \), and \( \mu^\mathcal{R}_X(\Phi)(x) = \bigoplus_{\varphi \in 2^X} \Phi(\varphi) \cdot \varphi(x) \).

• The nominal powerset monad \( \langle \mathcal{P}_n, \eta^{\mathcal{P}_n}, \mu^{\mathcal{P}_n} \rangle \) on the category of (finitely-supported) nominal sets is given by \( \mathcal{P}_nX = \{ A \subseteq X \mid A \text{ finitely supported} \} \), \( \pi.A := \{ \pi.a \mid a \in A \} \), and otherwise coincides with \( \mathcal{P} \) [45].
B Proofs

Lemma B.1 [22] Every algebra \( h : TB \to B \) for a set monad \( T \) induces a distributive law \( \lambda^h \) between \( T \) and \( F \) with \( FX = B \times X^A \) by \( \lambda^h_X := (h \times st) \circ (T \pi_1, T \pi_2) \).

Proof The statement is well-known [22], but a complete proof hard to find. The naturality of \( \lambda^h \) essentially follows from the naturality of the strength function. The equation involving the monad unit is a consequence of

\[
\begin{align*}
\pi_1 \circ (h \times st) & \circ (T \pi_1, T \pi_2) \circ \eta_{B \times X^A} \\
= h \circ T \pi_1 \circ \eta_{B \times X^A} & \quad \text{(Def. } \pi_1) \\
= h \circ \eta_B \circ \pi_1 & \quad \text{(Nat. } \eta) \\
= \pi_1 \circ (B \times \eta_X^A) & \quad (h \circ \eta_B = \text{id}_B, \text{Def. } \pi_1)
\end{align*}
\]

and

\[
\begin{align*}
\pi_2 \circ (h \times st) & \circ (T \pi_1, T \pi_2) \circ \eta_{B \times X^A} \\
= st \circ T \pi_2 \circ \eta_{B \times X^A} & \quad \text{(Def. } \pi_2) \\
= st \circ \eta_{X^A} \circ \pi_2 & \quad \text{(Nat. } \eta) \\
= \eta_X^A \circ \pi_2 & \quad (st \circ \eta_{X^A} = \eta_X^A) \\
= \pi_2 \circ (B \times \eta_X^A) & \quad \text{(Def. } \pi_2).
\end{align*}
\]

Similarly, the equation involving the monad multiplication is a consequence of

\[
\begin{align*}
\pi_1 \circ (B \times \mu_X^A) & \circ (h \times st) \circ (T \pi_1, T \pi_2) \circ T(h \times st) \circ T(T \pi_1, T \pi_2) \\
= h \circ Th \circ T^2 \pi_1 & \quad \text{(Def. } \pi_1) \\
= h \circ \mu_B \circ T^2 \pi_1 & \quad (h \circ Th = h \circ \mu_B) \\
= h \circ T \pi_1 \circ \mu_B \circ X^A & \quad \text{(Nat. } \mu) \\
= \pi_1 \circ (h \times st) \circ (T \pi_1, T \pi_2) \circ \mu_B \circ X^A & \quad \text{(Def. } \pi_1)
\end{align*}
\]

and

\[
\begin{align*}
\pi_2 \circ (B \times \mu_X^A) & \circ (h \times st) \circ (T \pi_1, T \pi_2) \circ T(h \times st) \circ T(T \pi_1, T \pi_2) \\
= \mu_X^A \circ st \circ T(st) \circ T^2 \pi_2 & \quad \text{(Def. } \pi_2) \\
= st \circ \mu_{X^A} \circ T^2 \pi_2 & \quad (\mu_X^A \circ st \circ T(st) = st \circ \mu_{X^A}) \\
= st \circ T \pi_2 \circ \mu_B \circ X^A & \quad \text{(Nat. } \mu) \\
= \pi_2 \circ (h \times st) \circ (T \pi_1, T \pi_2) \circ \mu_B \circ X^A & \quad \text{(Def. } \pi_2).
\end{align*}
\]

Lemma B.2 The morphism \( h^P : P2 \to 2 \) satisfying \( \varphi \mapsto \varphi(1) \) defines a \( P \)-algebra.

Proof On the one hand we find

\[
\begin{align*}
h^P \circ \mu^P_x(\Phi) & = \mu^P_{\Phi}((\Phi)(1)) \quad \text{(Def. } h^P) \\
& = \bigvee_{\varphi \in 2^2} \Phi(\varphi) \land \varphi(1) \quad \text{(Def. } \mu^P_x) \\
& = \bigvee_{\varphi \in (h^P)^{-1}(1)} \Phi(\varphi) \quad \text{(Def. } h^P, \text{ Def. } h^P) \\
& = P(h^P)(\Phi)(1) \quad \text{(Def. } P(h^P)) \\
& = h^P \circ P(h^P)(\Phi) \quad \text{(Def. } h^P)
\end{align*}
\]
and on the other hand we can deduce

\[ h^P \circ \eta^P_2(x) = \eta^P_2(x)(1) \]  \hspace{2cm} \text{(Def. } h^P) \]
\[ = [x = 1] \]  \hspace{2cm} \text{(Def. } \eta^P_2) \]
\[ = x \]  \hspace{2cm} \text{(Def. } [[]]) \].

\[ \square \]

**Lemma B.3** The morphism \( h^H \colon H2 \to 2 \) assigning \( \Phi \mapsto \Phi(\text{id}_2) \) defines a \( H \)-algebra.

**Proof** Since \( \eta^H_2(\text{id}_2)(\Phi) = \Phi(\text{id}_2) = h^H(\Phi) \) we find

\[ h^H \circ \mu^H_2(\Psi) = \mu^H_2(\Psi)(\text{id}_2) \]  \hspace{2cm} \text{(Def. } h^H) \]
\[ = \Psi(\eta^H_2(\text{id}_2)) \]  \hspace{2cm} \text{(Def. } \mu^H_2) \]
\[ = \Psi(\text{id}_2 \circ h^H) \]  \hspace{2cm} \text{(} \eta^H_2(\text{id}_2) = h^H \text{)} \]
\[ = h^H(\Psi)(\text{id}_2) \]  \hspace{2cm} \text{(Def. } H(\Psi)) \]
\[ = h^H \circ H(h^H)(\Psi) \]  \hspace{2cm} \text{(Def. } h^H). \]

We further can deduce

\[ h^H \circ \eta^H_X(x) = \eta^H_X(x)(\text{id}_2) \]  \hspace{2cm} \text{(Def. } h^H) \]
\[ = \text{id}_2(x) \]  \hspace{2cm} \text{(Def. } \eta^H_X) \]
\[ = x \]  \hspace{2cm} \text{(Def. } \text{id}_2). \]

\[ \square \]

**Lemma B.4** The morphism \( h^A : A2 \to 2 \) assigning \( \Phi \mapsto \Phi(\text{id}_2) \) defines a \( A \)-algebra.

**Proof** Analogous to the proof of Lemma B.3.

\[ \square \]

**Lemma B.5** The morphism \( h^R : R2 \to 2 \) satisfying \( \varphi \mapsto \varphi(1) \) defines a \( R \)-algebra.

**Proof** Analogous to the proof of Lemma B.2.

\[ \square \]

**Proposition 4.3** Let \( \langle X, h, k \rangle \) be a \( \lambda \)-bialgebra and let \( \langle Y, i, d \rangle \) be a generator for the \( T \)-algebra \( \langle X, h \rangle \). Then \( h \circ Ti : \exp_T(\langle Y, Fd \circ k \circ i \rangle) \to \langle X, h, k \rangle \) is a \( \lambda \)-bialgebra homomorphism.

**Proof** By definition we have \( \exp_T(\langle Y, Fd \circ k \circ i \rangle) = \langle TY, T\mu_Y, F\mu_Y \circ \lambda_{TY} \circ T(Fd \circ k \circ i) \rangle \). It is well-known that \( h \circ Ti \) is a homomorphism between the underlying \( T \)-algebra structures. It thus remains to show that it is a \( F \)-coalgebra homomorphism. The latter follows from the commutativity of the diagram below:

\[ \begin{array}{ccc}
TY & \xrightarrow{T_i} & TX \\
T_j & & \downarrow h \\
T X & & X \\
T_k & \downarrow & \\
T FX & \xrightarrow{id_{TFX}} & TFX \\
T Fd & \downarrow & \\
TFTY & \xrightarrow{TFTi} & TFTX \\
\lambda_{TY} & \downarrow & \\
F^2T Y & \xrightarrow{F^2T i} & F^2 TX \\
F \mu_Y & \downarrow & \ \ \\
FTY & \xrightarrow{FT i} & FT X \\
\end{array} \]

\[ \begin{array}{ccc}
\lambda_X & \downarrow & \\
F \mu_X & \downarrow & \ \ \\
FX & \xrightarrow{id_{FX}} & FX \\
\end{array} \]

\[ \square \]
**Proposition 4.4** Let \( \langle X, h, k \rangle \) be a \( \lambda \)-bialgebra and let \( \langle Y, i, d \rangle \) be a basis for the \( T \)-algebra \( \langle X, h \rangle \). Then \( h \circ Ti : \exp_T(\langle Y, Fd \circ k \circ i \rangle) \rightarrow \langle X, h, k \rangle \) is a \( \lambda \)-bialgebra isomorphism.

**Proof** By definition we have \( \exp_T(\langle Y, Fd \circ k \circ i \rangle) = (TY, \mu_Y, F\mu_Y \circ \lambda_{TY} \circ T(Fd \circ k \circ i)) \). From Proposition 4.3 we know that \( h \circ Ti \) is a \( \lambda \)-bialgebra homomorphism. By the definition of a basis, \( d \) is a two-sided inverse to \( h \circ Ti \) as ordinary morphism. It thus remains to show that \( d \) is a \( \lambda \)-bialgebra homomorphism. The diagram below on the left shows that it is a \( T \)-algebra homomorphism, and the diagram on the right below shows that it commutes with \( F \)-coalgebra structures:

\[
\begin{array}{cccc}
TX & \xrightarrow{Td} & T^2Y \\
\downarrow{id_{TX}} & & \downarrow{\tau^2_i} \\
TX & \xleftarrow{T_h} & T^2X \\
\downarrow{h} & & \downarrow{\mu_Y} \\
X & \xleftarrow{d} & TY \\
\end{array}
\]

\[
\begin{array}{cccc}
X & \xrightarrow{id_X} & X \\
\downarrow{d} & & \downarrow{h} \\
TY & \xrightarrow{T_h} & TX \\
\downarrow{T_k} & & \downarrow{\lambda_X} \\
TX & \xleftarrow{T^2} & \xrightarrow{T_d} TX \\
\downarrow{\tau_i} & & \downarrow{\lambda_{TY}} \\
TY & \xleftarrow{TY} & \xrightarrow{FTX} TFX \\
\downarrow{T_Fh} & & \downarrow{TFd} \\
FX & \xrightarrow{F_{\mu_Y}} & FTY
\end{array}
\]

\[\square\]

**Lemma B.6** The strength function \( st : \mathcal{P}_a(X^A) \rightarrow (\mathcal{P}_aX)^A \) satisfying \( st(\Phi)(a) = \mathcal{P}_a(ev_a)(\Phi) \) is equivariant.

**Proof** We first observe that for any \( a \in A, x \in X \) and \( \pi \in \text{Perm}(A) \) the mapping

\[
\{ \varphi \in X^A \mid \varphi(a) = x \} \rightarrow \{ \varphi \in X^A \mid \varphi(\pi^{-1}a) = \pi^{-1}x \} \quad \pi \mapsto \pi^{-1} \varphi
\]

defines a bijection with inverse assignment \( \varphi \mapsto \pi \varphi \). Note that the set 2 is equipped with the trivial action.

The statement thus follows from

\[
\begin{align*}
(\pi.st(\Phi))(a)(x) &= \pi.(st(\Phi)(\pi^{-1}.a))(x) \\
&= st(\Phi)(\pi^{-1}.a)(\pi^{-1}.x) \\
&= \mathcal{P}_a(ev_{\pi^{-1}.a})(\Phi)(\pi^{-1}.x) \\
&= \bigvee_{\varphi \in \mathcal{P}_a(ev_{\pi^{-1}}(x))} \Phi(\pi^{-1}\varphi) \\
&= \bigvee_{\varphi \in \mathcal{P}_a(ev_{\pi^{-1}}(x))} (\pi.\Phi)(\varphi) \\
&= \mathcal{P}_a(ev_a)(\pi.\Phi)(x) \\
&= st(\pi.\Phi)(a)(x) \\
&= \mathcal{P}_a(ev_a)(\pi.\Phi)(x) \\
&= \mathcal{P}_a(ev_a)(\pi.\Phi)(x)
\end{align*}
\]

\[\square\]

**Lemma B.7** Let \( \langle \mathcal{P}X, \mu_X^P, (\varepsilon, \delta) \rangle := \text{free}^P(\langle X, (\varepsilon, \delta) \rangle) \). Then \( \varepsilon(\varphi) = \bigvee_{y \in \varepsilon^{-1}(1)} \varphi(y) \) and \( \delta_a(\varphi)(x) = \bigvee_{y \in \mathcal{A}_a(x)} \varphi(y) \).

**Proof** The first equality is a consequence of

\[
\begin{align*}
\varepsilon(\varphi) &= \pi_1 \circ (h^P \times st) \circ \langle \mathcal{P}\pi_1, \mathcal{P}\pi_2 \rangle \circ \mathcal{P}(\langle \varepsilon, \delta \rangle)(\varphi) \\
&= h^P \circ \mathcal{P}(\varepsilon)(\varphi) \\
&= \mathcal{P}(\varepsilon)(\varphi)(1) \\
&= \bigvee_{y \in \varepsilon^{-1}(1)} \varphi(y) \\
&= \mathcal{P}(\varepsilon)(\varphi)(1)
\end{align*}
\]

\[\text{(Def. \varepsilon, Def. \pi_1, Def. h^P)}\]

\[\text{Def. \mathcal{P}(\varepsilon)}\]
For the second equality we observe
\[
\tilde{\delta}_a(\varphi)(x) = \tilde{\delta}(\varphi)(a)(x)
\]
(Def. \(\tilde{\delta}_a\))
\[
= \pi_2 \circ (h^P \times \text{st}) \circ (P\pi_1, P\pi_2) \circ P(\langle \varepsilon, \delta \rangle)(\varphi)(a)(x)
\]
(Def. \(\tilde{\delta}\))
\[
= \text{st} \circ P(\delta)(\varphi)(a)(x)
\]
(Def. \(\pi\))
\[
= P(\delta_a)(\varphi)(x)
\]
(Def. \(\delta_a\))
\[
= \bigvee_{y \in \delta_a^{-1}(a)} \varphi(y)
\]
(Def. \(P(\delta_a)\)).

\[\square\]

**Lemma B.8** Let \(\langle HX, \mu^X_H, \langle \varepsilon, \delta \rangle \rangle := \text{free}^X_{\lambda^H}(\langle X, \langle \varepsilon, \delta \rangle \rangle)\). Then \(\tilde{\varepsilon}(\Phi) = \Phi(\varepsilon)\) and \(\tilde{\delta}_a(\Phi)(\varphi) = \Phi(\varphi \circ \delta_a)\).

**Proof** The proof is analogous to the one of Lemma B.7. The first equality is a consequence of
\[
\tilde{\varepsilon}(\Phi) = H(\varepsilon)(\Phi)(\text{id}_2)
\]
(Cf. proof of Lemma B.7)
\[
= \Phi(\text{id}_2 \circ \varepsilon)
\]
(Def. \(H(\varepsilon)\))
\[
= \Phi(\varepsilon)
\]
(id \(\text{id}_2 \circ \varepsilon = \varepsilon\)).

For the second equality we observe
\[
\tilde{\delta}_a(\Phi)(\varphi) = H(\delta_a)(\Phi)(\varphi)
\]
(Cf. proof of Lemma B.7)
\[
= \Phi(\varphi \circ \delta_a)
\]
(Def. \(H(\delta_a)\)).

\[\square\]

**Lemma B.9** Let \(\langle AX, \mu^A_X, \langle \varepsilon, \delta \rangle \rangle := \text{free}^X_{\lambda^A}(\langle X, \langle \varepsilon, \delta \rangle \rangle)\). Then \(\tilde{\varepsilon}(\Phi) = \Phi(\varepsilon)\) and \(\tilde{\delta}_a(\Phi)(\varphi) = \Phi(\varphi \circ \delta_a)\).

**Proof** Analogous to the proof of Lemma B.8.

\[\square\]

**Lemma B.10** Let \(\langle RX, \mu^R_X, \langle \varepsilon, \delta \rangle \rangle := \text{free}^X_{\lambda^R}(\langle X, \langle \varepsilon, \delta \rangle \rangle)\). Then \(\tilde{\varepsilon}(\Phi) = \bigoplus_{y \in \varepsilon^{-1}(1)} \varphi(y)\) and \(\tilde{\delta}_a(\Phi)(x) = \bigoplus_{y \in \delta_a^{-1}(x)} \varphi(y)\).

**Proof** Analogous to the proof of Lemma B.7.

\[\square\]

**Lemma 5.2** \([27, 28]\) Let \(\alpha : \lambda^S \rightarrow \lambda^T\) be a distributive law homomorphism. Then \(\alpha\langle X, h, k \rangle := \langle X, h \circ \alpha_X, k \rangle\) and \(\alpha(f) := f\) defines a functor \(\alpha : \text{Bialg}(\lambda^S) \rightarrow \text{Bialg}(\lambda^T)\).

**Proof** The statement is well-known \([27, 28]\), but a complete proof difficult to find. We first show that the definition is well-defined on objects. The commutativity of the two diagrams below shows that \(\langle X, h \circ \alpha_X \rangle\) is a \(T\)-algebra:

To establish that \(\langle X, h \circ \alpha_X, k \rangle\) is a \(\lambda^T\)-bialgebra it thus remains to observe the commutativity of the diagram
Well-definedness on morphisms follows from the naturality of \(\alpha\), as seen on the right above. Compositionality follows immediately from the definition of \(\alpha\) on morphisms.

**Lemma B.11** Let \(\alpha : \lambda^S \to \lambda^T\) be a distributive law homomorphism. If \((\Omega, h, k)\) is the final \(\lambda^S\)-bialgebra, then \((\Omega, h \circ \alpha, k)\) is the final \(\lambda^T\)-bialgebra.

**Proof** It is well-known that if \((\Omega, h, k)\) is the final \(\lambda^S\)-bialgebra, then \((\Omega, k)\) is the final \(F\)-coalgebra and \(h : S\Omega \to \Omega\) is the unique homomorphism satisfying \(k \circ h = Fh \circ \lambda^S_{\Omega}\). Similarly, it is well-known that \((\Omega, h, k)\) is the final \(\lambda^T\)-bialgebra, where \(h : T\Omega \to \Omega\) is the unique homomorphism satisfying \(h \circ h = Fh \circ \lambda^T_{\Omega}\). The statement thus follows from uniqueness:

\[
\begin{array}{c}
T\Omega \xrightarrow{\alpha} S\Omega \xrightarrow{h} \Omega \\
T\Omega \xrightarrow{T\pi_1} S\Omega \xrightarrow{S\pi_1} \Omega \\
TF\Omega \xrightarrow{\alpha_{F\Omega}} SF\Omega \xrightarrow{h} \Omega \\
FT\Omega \xrightarrow{\lambda^S_{\Omega}} F\Omega \xrightarrow{Fh} \Omega \\
\end{array}
\]

**Corollary 5.3** Let \(\alpha : \lambda^S \to \lambda^T\) be a homomorphism between distributive laws and \((X, h, k)\) a \(\lambda^S\)-bialgebra. If \((Y, i, t)\) is a generator for the \(T\)-algebra \((X, h \circ \alpha_X)\), then \((h \circ \alpha_X) \circ Ti : \exp_T((Y, i \circ k \circ i)) \to (X, h \circ \alpha_X, k)\) is a \(\lambda^T\)-bialgebra homomorphism.

**Proof** By Lemma 5.2 the tuple \((X, h \circ \alpha_X, k)\) constitutes a \(\lambda^T\)-bialgebra. The statement thus follows from Proposition 4.3.

**Lemma 5.4** Let \(\alpha : T \Rightarrow S\) be a natural transformation satisfying \(h^S \circ \alpha_B = h^T\), then \(\lambda^S \circ \alpha_F = F\alpha \circ \lambda^T\).

**Proof** We need to establish the commutativity of the following diagram:

\[
\begin{array}{c}
(TX)A \times B \xrightarrow{(\alpha_A \times B)} S(XA \times B) \\
\downarrow \quad \downarrow \\
T(A \times B) \xrightarrow{\alpha_A \times B} S(A \times B) \\
\end{array}
\]
The commutativity of the top square is a consequence of the naturality of $\alpha$. Similarly, the commutativity of the bottom square follows from the assumption and the naturality of $\alpha$,

$$
\text{st} \circ \alpha_X (U)(a) = S(\text{ev}_a) \circ \alpha_X (U) \quad \text{(Def. st)} \\
= \alpha_X \circ T(\text{ev}_a)(U) \quad \text{(Nat. } \alpha) \\
= \alpha^A_X \circ \text{st}(U)(a) \quad \text{(Def. st)}.
$$

\[ \blacksquare \]

**Proposition 5.5** Any algebra $h : T2 \to 2$ over a set monad $T$ induces a homomorphism $\alpha^h : \lambda^H \to \lambda^b$ between distributive laws by $\alpha^h_X := h^2 X \circ \text{st} \circ T(\eta^H_X)$.

**Proof** It is well-known that the strength operation is natural and satisfies the equalities $(\eta^T_A)^B = \text{st} \circ \eta^T_{AB}$ and $\text{st} \circ \mu_{AB} = \mu^B \circ \text{st} \circ T\text{st}$. It is also not hard to see that for functions $f : A \to B$ and $g : C \to D$ it holds $f^D \circ A^g = B^g \circ f^C$. We write $f^*$ for $A^f$ and $f_*$ for $f^A$, and omit components of natural transformations for readability. The naturality of $\alpha^h_X$ is a consequence of:

\[
\begin{array}{c}
T X \xrightarrow{T\eta^H_X} T(2^X) \xrightarrow{\text{st}} T(2)^2^X \xrightarrow{h_*} 2^2^X \\
T Y \xrightarrow{T\eta^H_Y} T(2^Y) \xrightarrow{\text{st}} T(2)^2^Y \xrightarrow{h_*} 2^2^Y.
\end{array}
\]

Using the equality $2^{\eta^H_X} \circ \eta^H_{2^2^X} = \text{id}_{2^2^X}$, the equation involving the monad multiplications is seen as follows:

\[
\begin{array}{c}
T(2^2^X) \xrightarrow{T(\eta^H_2)} T(2^2^2^X) \xrightarrow{\text{st}} T(2)^2^2^2^X \xrightarrow{h_*} 2^2^2^2^X \\
T(h_*) \\
T(T(2)^2^X) \xrightarrow{\text{st}} T^2(2)^2^X \xrightarrow{T(h)_*} T(2)^2^X \xrightarrow{(\eta^H_2)_*}. \\
T^{\mu^T}(X) \xrightarrow{\mu^T} X \xrightarrow{h_*} 2^X \\
T(X) \xrightarrow{T(\eta^h_X)} T(2^X) \xrightarrow{\text{st}} T(2)^2^X \xrightarrow{h_*} 2^2^X.
\end{array}
\]
The equation involving the monad units is established by:

\[
\begin{array}{c}
X \xrightarrow{T} TX \\
\downarrow \eta_T \\
\eta_H \downarrow \\
2^X \xrightarrow{h_*} (2^X)^2 \\
\end{array}
\]

To show that the equation involving the distributive laws holds, we use Lemma 5.4. That is, we note that for any \( f \) it holds \( f \circ ev_a = ev_a \circ f_* \), and moreover, \( h^H = ev_{id_2} \), before establishing:

\[
\begin{array}{c}
T(2) \xrightarrow{T(\eta^H)} T(2^2) \\
\downarrow st \\
\downarrow \\
(2^2)^2 \xrightarrow{h_*} 2^2 \\
\end{array}
\]

Corollary 5.6 Let \( \alpha_X : \mathcal{P}X \rightarrow \mathcal{H}X \) satisfy \( \alpha_X(\psi)(\varphi) = \bigvee_{x \in X} \varphi(x) \wedge \psi(x) \), then \( \alpha \) constitutes a distributive law homomorphism \( \lambda^H \rightarrow \lambda^P \).

Proof We show that \( \alpha^{h^P} = \alpha \), the statement then follows from Proposition 5.5. We calculate

\[
\begin{align*}
\alpha_X^{h^P}(\varphi)(\psi) &= (h^P)^2 \circ \eta^H \circ \eta^P(\varphi)(\psi) \\
&= st \circ \mathcal{P}(\eta^H^X)(\varphi)(1) \\
&= \mathcal{P}(\psi)(\varphi)(1) \\
&= \bigvee_{x \in \psi^{-1}(1)} \varphi(x) \\
&= \bigvee_{x \in X} \varphi(x) \wedge \psi(x) \\
&= \alpha_X(\varphi)(\psi) \\
\end{align*}
\]

Lemma 5.7 Let \( \alpha_X : \mathcal{P}X \rightarrow \mathcal{H}X \) satisfy \( \alpha_X(\varphi)(\psi) = \bigvee_{x \in X} \varphi(x) \wedge \psi(x) \). If \( B = (X, h) \) is a \( \mathcal{H} \)-algebra, then \( \langle \mathcal{A}(B), i, d \rangle \) with \( i(a) = a \) and \( d(x) = \{ a \in \mathcal{A}(B) \mid a \leq x \} \) is a basis for the \( \mathcal{P} \)-algebra \( \langle X, h \circ \alpha_X \rangle \).

Proof Let \( K : \text{Set}^{op} \rightarrow \text{Alg}(\mathcal{H}) \) denote the comparison functor with \( K(X) = \langle \mathcal{P}X, 2^{h^X} \rangle \) induced by the self-dual contravariant powerset adjunction. It is well-known that \( K \) has a quasi-inverse, namely the functor
Let \( \eta \): \( A \to \mathcal{H} \) denote the monad \( A \), analogous to the proof of Corollary 5.6.

**Proof** We observe that the \( \eta \): \( 2^X, \subseteq \to \langle 2, \subseteq \rangle \) is monotone for all \( \varphi \in 2^X \). Since the monotone neighbourhood monad \( A \) and the neighbourhood monad \( H \) only differ on objects, the result follows from Corollary 5.6. □

**Corollary 5.8** Let \( \alpha \): \( \mathcal{P}X \to \mathcal{A}X \) satisfy \( \alpha(\varphi)(\psi) = \bigvee_{x \in X} \varphi(x) \wedge \psi(x) \), then \( \alpha \) constitutes a distributive law homomorphism \( \alpha : \lambda^A \to \lambda^P \).

**Proof** Analogous to the proof of Corollary 5.6. □

**Proposition 6.1** Let \( \alpha : \lambda^S \to \lambda^T \) be a distributive law homomorphism. Then \( \alpha \): \( TX \to SX \) underlies a natural transformation \( \alpha : \exp_T \Rightarrow \alpha \circ \exp_S \circ \text{ext} \) between functors of type \( \text{Coalg}(FT) \to \text{Bialg}(\lambda^T) \).

**Proof** Given a \( T \)-succinct automaton \( X = \langle X, k \rangle \) the definitions imply

\[
\text{exp}_T(X) = \langle TX, \mu^T_X, F\mu^T_X \circ \lambda^T_X \circ Tk \rangle \quad \alpha \circ \text{exp}_S \circ \text{ext}(X) = \langle SX, \mu^S_X \circ \alpha_{SX}, F\mu^S_X \circ \lambda^S_{SX} \circ SF\alpha_X \circ Sk \rangle.
\]

By the definition of distributive law homomorphisms, the morphism \( \alpha \) commutes with the underlying \( F \)-coalgebra structures. Its commutativity with the underlying \( F \)-coalgebra structures follows from:

\[
\begin{align*}
\alpha_X : TX &\xrightarrow{Tk} TFTX \xrightarrow{T\alpha_X} FT^2X \xrightarrow{F\mu^T_X} FTX \\
\alpha_{FTX} : TFX &\xrightarrow{T\alpha_X} FT^2X \xrightarrow{F\mu^T_X} FTX \\
\alpha_{FSX} : FTX &\xrightarrow{F\alpha_X} F^2SX \xrightarrow{F\mu^S_X} FSX
\end{align*}
\]

For above we use the naturality of \( \alpha \) and \( \lambda^T \), and the definition of a distributivity law homomorphism. The naturality of \( \alpha \) as natural transformation \( \alpha : \exp_T \Rightarrow \alpha \circ \exp_S \circ \text{ext} \) follows immediately from the naturality of \( \alpha \) as natural transformation \( \alpha : T \Rightarrow S \). □

**Lemma 6.3** Let \( \alpha : \lambda^S \to \lambda^T \) be a distributive law homomorphism and \( \langle X, h, k \rangle \) a \( \lambda^S \)-bialgebra. If \( \langle Y, i, d \rangle \) is a generator for \( \langle X, h \circ \alpha_X \rangle \), then \( \langle Y, Fd \circ k \circ i \rangle \) is \( \alpha \)-closed.
Theorem 6.4 Given a language \( \mathcal{L} \in \Omega \) such that there exists a minimal pointed \( \lambda^S \)-bialgebra \( M \) accepting \( \mathcal{L} \) and the underlying algebra of \( \alpha(M) \) admits a size-minimal generator, there exists a pointed \( \alpha \)-closed \( T \)-sucint automaton \( X \) accepting \( \mathcal{L} \) such that:

- for any pointed \( \alpha \)-closed \( T \)-sucint automaton \( Y \) accepting \( \mathcal{L} \) we have that \( \text{im}(\text{obs}_{\exp_T(X)}) \subseteq \text{im}(\text{obs}_{\exp_T(Y)}) \);
- if \( \text{im}(\text{obs}_{\exp_T(X)}) = \text{im}(\text{obs}_{\exp_T(Y)}) \), then \( |X| \leq |Y| \), where \( X \) and \( Y \) are the carriers of \( X \) and \( Y \), respectively.

Proof We use a similar notation as in the proof of Lemma 6.3. Let \( \mathbb{G} = (X, i, d) \) be the size-minimal generator for the underlying algebra of \( \alpha(M) \), which we assume to be \( x \)-pointed. We define a \( d(x) \)-pointed \( T \)-sucint automaton \( \mathcal{X} := \text{gen}(\alpha(M), \mathbb{G}) \), which by the existence of the pointed \( \lambda^T \)-algebra homomorphism \( \nu^T : \exp(X) \to \alpha(M) \) accepts the language accepted by \( \alpha(M) \). Because \( \alpha \) only modifies the algebraic part of a bialgebra and the final bialgebra homomorphism is induced by the underlying final coalgebra homomorphism, the language accepted by \( \alpha(M) \) is the language \( \mathcal{L} \) accepted by \( M \). From Lemma 6.3 it follows that \( X \) is \( \alpha \)-closed.

Consider any pointed \( \alpha \)-closed \( T \)-sucint automaton \( Y \) accepting \( \mathcal{L} \). Then by minimality of \( M \) there exists an injective \( \lambda^T \)-bialgebra homomorphism \( j : M \to \text{im}(\text{obs}_{\exp_T(Y)}) \), which is also a \( \lambda^T \)-biyomorphism \( j : \alpha(M) \to \text{im}(\text{obs}_{\exp_T(Y)}) \)), because the functor \( \alpha \) is the identity on morphisms, and only modifies the algebraic part of a bialgebra. Since \( \mathcal{Y} \) is \( \alpha \)-closed, the codomain of \( j \) is isomorphic to \( \text{im}(\text{obs}_{\exp_T(Y)}) \). Moreover, \( \alpha(M) \) is isomorphic to \( \text{im}(\text{obs}_{\exp_T(X)}) \) by the existence of a surjective homomorphism \( \exp_T(X) \to \alpha(M) \). The resulting \( \lambda^T \)-bialgebra homomorphism \( \text{im}(\text{obs}_{\exp_T(X)}) \to \text{im}(\text{obs}_{\exp_T(Y)}) \) commutes with observability maps and thus must be an inclusion map, so \( \text{im}(\text{obs}_{\exp_T(X)}) \subseteq \text{im}(\text{obs}_{\exp_T(Y)}) \).

Suppose \( \text{im}(\text{obs}_{\exp_T(X)}) = \text{im}(\text{obs}_{\exp_T(Y)}) \), which implies that \( j \) is an isomorphism. Then there exists a surjective \( \lambda^T \)-bialgebra homomorphism \( \exp_T(Y) \to \alpha(M) \), which means that \( Y \) forms the carrier of a generator for the underlying algebra of \( \alpha(M) \). By the size-minimality of \( \mathbb{G} \) we thus obtain \( |X| \leq |Y| \).

Lemma B.12 Any basis for a \( \mathcal{R} \)-algebra is a size-minimal generator.

Proof Let \( (Y, i, d) \) be a basis for a \( \mathcal{R} \)-algebra \( X \), then \( d : X \to \mathcal{R}Y \) is a bijection with inverse \( h \circ \mathcal{R}i \). Let \( (Y', i', d') \) be any other generator for \( X \), then \( h \circ \mathcal{R}i' : \mathcal{R}Y' \to \mathcal{R}Y \) is a surjection, which shows that \( \mathcal{R}^{-1} = 2^{-} \).

Lemma B.13 Any basis for a \( \mathcal{P} \)-algebra is a size-minimal generator.

Proof Analogous to the proof of Lemma B.12.

Corollary B.14 Let \( \alpha_X : PX \to HX \) satisfy \( \alpha_X(\varphi)(\psi) = \bigvee_{x \in X} \varphi(x) \land \psi(x) \). If \( B = (X, h) \) is a \( \mathcal{H} \)-algebra, then \( \langle \text{At}(B), i', d \rangle \) with \( i(a) = a \) and \( d(x) = \{ a \in \text{At}(B) \mid a \leq x \} \) is a size-minimal generator for \( (X, h \circ \alpha_X) \).
**Proof** By Lemma 5.7 $\langle \text{At}(B), i, d \rangle$ is a basis for $\langle \mathcal{X}, h \circ \alpha_X \rangle$, which by Lemma B.13 implies size-minimality. □

**Lemma B.15** For any finite $\mathcal{P}$-algebra $L = \langle \mathcal{X}, h \rangle$ the join-irreducibles $\langle J(L), i, d \rangle$ with $i(y) = y$ and $d(x) = \{y \in J(L) \mid y \leq x\}$ constitute a size-minimal generator.

**Proof** Since $L$ is finite, it satisfies the descending chain condition (DCC), which in turn can be used to show that the join-irreducibles constitute a generator as follows.

Assume there exists some $x \in X$ with $x \neq i^2(d(x))$. We build an infinite sequence $(a_n)$ with $a_i > a_{i+1}$ and $a_i \neq i^2(d(a_i))$, which contradicts the DCC. For the base case we define $a_0 := x$. For any $x \in X$, the property $x \in J(L)$ immediately implies $x = i^2(d(x))$. Thus we can assume $a_i \notin J(L)$. In consequence we have $a_i = y \lor z$ for $y, z \neq a_i$, i.e. $a_i > y$ and $a_i > z$. Assume $y = i^2(d(y))$ and $z = i^2(d(z))$. Then

$$i^2(d(a_i)) \leq a_i = y \lor z = i^2(d(y)) \lor i^2(d(z)) = i^2(d(y) \lor d(z)) \leq i^2(d(a_i)).$$

It thus follows $a_i = i^2(d(a_i))$, which is a contradiction. Hence, w.l.o.g. assume $y \neq i^2(d(y))$ and define $a_{i+1} := y$.

Let $(Y, i', d')$ be an arbitrary generator for $L$. For any $a \in J(L)$ we have $a = \bigvee_{y \in d'(a)} i'(y)$. By the definition of join-irreducibles there exists at least one $y_0 \in d(a)$ such that $i'(y_0) = a$. One can thus define a function $f : J(L) \to Y$ with $f(a) = y_0$. It is not hard to see that $f$ is injective, which implies $|J(L)| \leq |Y|$. □

**Lemma B.16** Let $X = \langle \mathcal{X}, h, k \rangle$ be an observable $\lambda^S$-bialgebra and $G$ a generator for $\langle \mathcal{X}, h \circ \alpha_X \rangle$, then $\text{im}(\text{obs}_{\exp_T \circ (\text{gen}(\alpha(X), G))}) \simeq X$.

**Proof** By Proposition 4.3 there exists a surjective $\lambda^T$-bialgebra homomorphism of type $\exp_T \circ (\text{gen}(\alpha(X), G)) \to \alpha(X)$. Since the final $\lambda^T$-bialgebra homomorphism is induced by the underlying final $F$-coalgebra homomorphism and $\alpha(X) = \langle \mathcal{X}, h \circ \alpha_X, k \rangle$, it holds $\text{obs}_{\alpha(X)} = \text{obs}_X$. The statement follows from the uniqueness of epi-mono factorizations and the definition of $\alpha(X)$:

![Diag: exp(gen(X,G)) -> alpha(X) -> im(obs_{exp_T(gen(X,G))}) -> Omega](image)

□

**Lemma B.17** Let $X$ be a $T$-succinct automaton, then $\text{obs}_{\alpha_X}^\dagger = \text{obs}_{\text{ext}(X)}^\dagger$.

**Proof** Since by Proposition 6.1 the morphism $\alpha_X : \exp_T(X) \to \alpha(\exp_T(\text{ext}(X)))$ is a $\lambda^T$-bialgebra homomorphism, we have by uniqueness $\text{obs}_{\alpha(\exp_T(\text{ext}(X)))} \circ \alpha_X = \text{obs}_{\exp_T(X)}$. Since any final bialgebra homomorphism is induced by the underlying final $F$-coalgebra homomorphism it holds $\text{obs}_{\alpha(\exp_T(\text{ext}(X)))} = \text{obs}_{\exp(\text{ext}(X))}$, which thus implies

$$\text{obs}_{\exp_T(\text{ext}(X))} \circ \alpha_X = \text{obs}_{\exp_T(X)}. \quad (B.2)$$

The statement follows from

$$\text{obs}_{\alpha_X}^\dagger = \text{obs}_{\exp_T(X)} \circ \eta^X_T \quad \text{(Def. obs}_{\alpha_X}^\dagger)$$

$$= \text{obs}_{\exp_T(\text{ext}(X))} \circ \alpha_X \circ \eta^X_T \quad \text{(B.2)}$$

$$= \text{obs}_{\exp_T(\text{ext}(X))} \circ \eta^S_X \quad \text{(Def. distr. law hom.)}$$

$$= \text{obs}_{\text{ext}(X)}^\dagger \quad \text{(Def. obs}_{\text{ext}(X)}^\dagger).$$

□

**Lemma 6.5** Let $\alpha : \lambda^S \to \lambda^T$ be a distributive law homomorphism and $\langle \Omega, h, k \rangle$ the final $\lambda^S$-bialgebra. For any $T$-succinct automaton $X$ it holds $\text{im}(\text{obs}_{\alpha_X}) = \text{im}(h \circ \alpha \circ T(\text{obs}_{\alpha_X}^\dagger))$ and $\text{im}(\text{obs}_{\alpha(\exp_T(\text{ext}(X)))}) = \text{im}(h \circ S(\text{obs}_{\alpha_X}^\dagger))$. 24
Proof By Lemma B.11 $\langle \Omega, h \circ \alpha, k \rangle$ is the final $\lambda^T$-bialgebra. The first statement follows from

$$\text{obs}_{\exp T(\mathcal{X})} = \text{obs}_{\exp T(\mathcal{X})} \circ \mu_{\mathcal{T}X} \circ T(\eta_{\mathcal{T}X}) = h \circ \alpha \circ T(\text{obs}_{\exp T(\mathcal{X})}) \circ T(\eta_{\mathcal{T}X}) = h \circ \alpha \circ T(\text{obs}_{\mathcal{T}X}^T).$$

(Def. monad $T$)

(Algebra hom. $\text{obs}_{\exp T(\mathcal{X})}$)

(Def. $\text{obs}_{\mathcal{T}X}^T$).

Similarly one shows that $\text{obs}_{\exp T(\mathcal{X})} = h \circ S(\text{obs}_{\exp T(\mathcal{X})})$. Since any final bialgebra homomorphism is induced by the underlying final $F$-coalgebra homomorphism, it thus follows

$$\text{im}(\text{obs}_{\alpha(\exp T(\mathcal{X})))} = \text{im}(\text{obs}_{\exp T(\mathcal{X})))} \quad \text{by the underlying final $F$-coalgebra homomorphism, it thus follows}$$

$$\text{im}(\text{obs}_{\alpha(\exp T(\mathcal{X})))}) = \text{im}(\text{obs}_{\mathcal{T}X}^T)$$

(Def. $\text{im}(\text{obs}_{\alpha(\exp T(\mathcal{X})))})$)

(Lemma B.17).

Corollary B.18 Let $\alpha : \mathcal{P}X \rightarrow \mathcal{H}X$ satisfy $\alpha_X(\varphi)(\psi) = \bigvee_{\chi \in X} \varphi(x) \land \psi(x)$. For any unpointed non-deterministic automaton $\mathcal{X}$ it holds:

- $\text{im}(\text{obs}_{\exp T(\mathcal{X})))} = \text{im}(\text{obs}_{\mathcal{T}X}^C)_\text{CSL}$;
- $\text{im}(\text{obs}_{\alpha(\exp T(\mathcal{X})))}) = \text{im}(\text{obs}_{\mathcal{T}X}^T)_\text{CABA}$.

Proof The final $\lambda^T$-bialgebra is given by $\langle 2^{A^T}, 2^{H^T}_X, \langle \varepsilon, \delta \rangle \rangle$. In the proof of Lemma 5.7 it was shown that $2^{\lambda^T} \circ \alpha_{\mathcal{P}X} = \mu_{\mathcal{T}X}^T$. Thus it follows

$$\text{im}(\text{obs}_{\exp T(\mathcal{X})))} = \text{im}(2^{\lambda^T} \circ \alpha_{2^A^T} \circ \mathcal{P}(\text{obs}_{\mathcal{T}X}^T))$$

(Lemma 6.5)

$$= \text{im}(\mu_{\mathcal{T}X}^T \circ \mathcal{P}(\text{obs}_{\mathcal{T}X}^T))$$

($2^{\lambda^T} \circ \alpha_{\mathcal{P}X} = \mu_{\mathcal{T}X}^T$)

$$= \{ \bigcup_{\mathcal{U} \in \mathcal{U}} \text{obs}_{\mathcal{T}X}^T(x) \mid x \in X \}$$

(Def. $\mathcal{P}(\cdot), \mu_{\mathcal{T}X}^T$)

$$= \{ \text{obs}_{\mathcal{T}X}^T(x) \mid x \in X \}$$

(Def. $\langle \cdot \rangle_\text{CSL}$).

Similarly one computes

$$\text{im}(\text{obs}_{\alpha(\exp T(\mathcal{X})))}) = \text{im}(2^\mathcal{H}_X \circ \mathcal{H}(\text{obs}_{\mathcal{T}X}^T))$$

(Lemma 6.5)

$$= \{ \{ w \in A^T \mid \{ x \in X \mid \text{obs}_{\mathcal{T}X}^T(x)(w) = 1 \} \in \Phi \} \mid \Phi \subseteq 2^X \}$$

(Def. $2^{\lambda^T}, \mathcal{H}(\cdot)$)

$$= \{ \bigcup_{\Phi \in \mathcal{F}} \bigcap_{x \in \mathcal{F}} \text{obs}_{\mathcal{T}X}^T(x) \cap \bigcap_{x \in \Phi} \text{obs}_{\mathcal{T}X}^T(x)^\Phi \mid \Phi \subseteq 2^X \}$$

(Set equality)

$$= \{ \text{obs}_{\mathcal{T}X}^T(x) \mid x \in X \}$$

(Def. $\langle \cdot \rangle_\text{CABA}$).

Corollary B.19 Let $\alpha : \mathcal{R}X \rightarrow \mathcal{A}X$ satisfy $\alpha_X(\varphi)(\psi) = \bigvee_{\varphi \in X} \varphi(x) \land \psi(x)$. For any unpointed non-deterministic automaton $\mathcal{X}$ it holds:

- $\text{im}(\text{obs}_{\exp T(\mathcal{X})))} = \text{im}(\text{obs}_{\mathcal{T}X}^C)_\text{CSL}$;
- $\text{im}(\text{obs}_{\alpha(\exp T(\mathcal{X})))}) = \text{im}(\text{obs}_{\mathcal{T}X}^T)_\text{CDL}$.

Proof Analogous to the proof of Corollary B.18.

Corollary B.20 Let $\alpha : \mathcal{R}X \rightarrow \mathcal{H}X$ satisfy $\alpha_X(\varphi)(\psi) = \bigoplus_{\varphi \in X} \varphi(x) \cdot \psi(x)$. For any unpointed $\mathbb{Z}_2$-weighted automaton $\mathcal{X}$ it holds:
Theorem B.6 The canonical RFSA for $L$ is size-minimal among non-deterministic automata $Y$ accepting $L$ with $\im(\text{obs}_{\text{exp}_Y}(Y))_{\text{CSL}} \leq \text{Der}(L)_{\text{CSL}}$.

Proof By Lemma B.21 the morphism $h^P : P2 \to 2$ with $h^P(\varphi) = \varphi(1)$ is a $P$-algebra. As shown in Lemma B.1, it can be used to derive a canonical distributive law $\lambda^P$. It is not hard to see that the minimal pointed $\lambda^P$-bialgebra $M$ accepting $L$ exists and that its underlying state space is given by the finite complete join-semi lattice $\text{Der}(L)_{\text{CSL}}$. By Lemma B.15 the join-irreducibles for $M$ constitute a size-minimal generator $G$. By definition, the canonical RFSA for $L$ is given by $X := \text{gen}(M, G)$. From Lemma B.16 it follows that $\im(\text{obs}_{\text{exp}_p}(Y)) \simeq \text{Der}(L)_{\text{CSL}}$. As seen in e.g. Corollary B.18, one has $\im(\text{obs}_{\text{exp}_p}(Y)) = \im(\text{obs}_Y^{\text{CSL}})$ for any NFA $Y$. By choosing $\alpha$ as the identity, which implies $\alpha$-closedness for any NFA, the statement thus follows from Theorem 6.4.

Corollary B.7 The minimal xor automaton for $L$ is size-minimal among $\mathbb{Z}_2$-weighted automata accepting $L$.

Proof Analogous to Corollary B.6 one can show that the minimal xor automaton for $L \subseteq A^*$ is size-minimal among all $\mathbb{Z}_2$-weighted automata $Y$ accepting $L$ such that $\im(\text{obs}_Y^{\text{CSL}}_{\mathbb{Z}_2\text{-Vect}}) \subseteq \text{Der}(L)_{\mathbb{Z}_2\text{-Vect}}$. Specific to this case are Lemma B.5 and Lemma B.12. It remains to observe that for any $\mathbb{Z}_2$-weighted automaton $X$, one can find an equivalent $\mathbb{Z}_2$-weighted automaton $Y$ with a state space of size not greater than the one of $X$, such that above inclusion holds. The state space of $Y$ can be chosen as a basis for the underlying vector space of the epi-mono factorisation of the reachability map $X(A^*) \to \im(\text{obs}_X^{\text{CSL}}_{\mathbb{Z}_2\text{-Vect}})$.

Corollary B.8 The automaton for $L$ is size-minimal among non-deterministic automata $Y$ accepting $L$ with $\im(\text{obs}_Y^{\text{CSL}})_{\text{CSL}} | \im(\text{obs}_Y^{\text{CSL}})_{\text{CABA}}$.

Proof Let $\alpha : \lambda^H \to \lambda^P$ be the distributive law homomorphism introduced in Corollary 5.6. As shown in Corollary B.18, the equality in above claim captures $\alpha$-closedness of $Y$. By construction there exists a CSL-epimorphism $\text{obs}_{\text{exp}_p}(Y) : PY \to \im(\text{obs}_Y^{\text{CSL}})$, which can be used to turn $Y$ into a $P$-algebra generator for $B := \im(\text{obs}_Y^{\text{CSL}})_{\text{CABA}} = \im(\text{obs}_Y^{\text{CSL}})$. As for CABA join-irreducibles and atoms coincide, the size-minimality of join-irreducibles in Lemma B.15 thus implies $|\text{At}(B)| \leq |Y|$. Since $\text{Der}(L)_{\text{CABA}}$ underlies the minimal $\alpha$-closed pointed $\lambda^H$-bialgebra accepting $L$, we have $\text{Der}(L)_{\text{CABA}} \subseteq B$ by Theorem 6.4, where we use Corollary B.14 and Lemma B.16. By Corollary B.22, the former implies $|\text{At}(\text{Der}(L)_{\text{CABA}})| \leq |\text{At}(B)|$. Consequently we can deduce $|\text{At}(\text{Der}(L)_{\text{CABA}})| \leq |Y|$, which shows the claim.
Corollary 6.9 The distromaton for $L$ is size-minimal among non-deterministic automata $Y$ accepting $L$ with $\text{im}(\text{obs}_{Y}^\dagger) = \text{im}(\text{obs}_{Y}^\dagger)^\text{CDL}$.

Proof Analogous to the proof of Corollary 6.8. Specific to this case are Lemma B.2, Lemma B.4, Corollary 5.8, Lemma B.15, Lemma B.21, and Corollary B.19.

Corollary 6.10 The minimal xor-CABA automaton for $L$ is size-minimal among $\mathbb{Z}_2$-weighted automata $Y$ accepting $L$ with $\text{im}(\text{obs}_{Y}^\dagger)^{\mathbb{Z}_2-\text{Vect}} = \text{im}(\text{obs}_{Y}^\dagger)^{\text{CABA}}$.

Proof Analogous to the proof of Corollary 6.8. Specific to this case are Lemma B.3, Lemma B.5, Corollary 5.9, Lemma B.12, Corollary B.20 and the observation that if $A \subseteq B$ is a sub-vector space of a finite vector space $B$, then $\text{dim}(A) \leq \text{dim}(B)$.

Corollary 6.11

• If $\text{Der}(L)^{\mathbb{Z}_2-\text{Vect}} = \text{Der}(L)^{\text{CABA}}$, then the minimal xor automaton and the minimal xor-CABA automaton for $L$ are of the same size.

• If $\text{Der}(L)^{\text{CSL}} = \text{Der}(L)^{\text{CDL}}$, then the canonical RFSA and the distromaton for $L$ are of the same size.

• If $\text{Der}(L)^{\text{CSL}} = \text{Der}(L)^{\text{CABA}}$, then the canonical RFSA and the átomaton for $L$ are of the same size.

Proof

• By Corollary 6.7 the minimal xor automaton $X$ is of size not greater than the minimal xor-CABA automaton $Y$. Conversely, we find

$$\text{Der}(L)^{\text{CABA}} = \text{Der}(L)^{\mathbb{Z}_2-\text{Vect}} \quad \text{(Assumption)}$$

$$= \text{im}(\text{obs}_{\text{exp}_X(Y)}) \quad \text{(Lemma B.16)}$$

$$= \text{im}(\text{obs}_{X}^{\mathbb{Z}_2-\text{Vect}}) \quad \text{(Corollary B.20)},$$

which can be used to show $\text{im}(\text{obs}_{X}^{\mathbb{Z}_2-\text{Vect}}) = \text{im}(\text{obs}_{X}^{\text{CABA}})$. By Corollary 6.10 the latter implies that $Y$ is of size not greater than $X$, which shows the claim.

• Let $X$ denote the canonical RFSA and $Y$ the distromaton. On the one hand we find

$$\text{Der}(L)^{\text{CSL}} = \text{Der}(L)^{\text{CDL}} \quad \text{(Assumption)}$$

$$= \text{im}(\text{obs}_{\text{exp}_X(Y)}) \quad \text{(Lemma B.16)}$$

$$= \text{im}(\text{obs}_{Y}^{\text{CSL}}) \quad \text{(Corollary B.19)},$$

which by Corollary 6.6 implies that $X$ is of size not greater than $Y$. Conversely, we establish the equality

$$\text{Der}(L)^{\text{CDL}} = \text{Der}(L)^{\text{CSL}} \quad \text{(Assumption)}$$

$$= \text{im}(\text{obs}_{\text{exp}_X(Y)}) \quad \text{(Lemma B.16)}$$

$$= \text{im}(\text{obs}_{X}^{\text{CDL}}) \quad \text{(Corollary B.19)},$$

which can be used to show $\text{im}(\text{obs}_{X}^{\text{CSL}}) = \text{im}(\text{obs}_{X}^{\text{CDL}})$. By Corollary 6.9 the latter implies that $Y$ is of size not greater than $X$, which shows the claim.

• The proof for the átomaton is analogous to the proof for the distromaton in the previous point.

\[\square\]