

Pointwise Extensions of GSOS-Defined Operations

HELLE HVID HANSEN¹ and BARTEK KLIN^{2†}

¹ *Technische Universiteit Eindhoven and Centrum Wiskunde & Informatica, P.O.Box 513, 5300 MB Eindhoven, The Netherlands.*

Email: h.h.hansen@tue.nl.

² *University of Cambridge and University of Warsaw, 15 JJ Thomson Avenue, Cambridge CB3 0FD, UK.*

Email: klin@mimuw.edu.pl.

Received 26 September 2010

Final coalgebras capture system behaviours such as streams, infinite trees and processes. Algebraic operations on a final coalgebra can be defined by distributive laws (of a syntax functor Σ over a behaviour functor F). Such distributive laws correspond to abstract specification formats. One such format is a generalisation of the GSOS rules known from structural operational semantics of processes. We show that given an abstract GSOS specification ρ that defines operations σ on a final F -coalgebra, we can systematically construct a GSOS specification $\bar{\rho}$ that defines the pointwise extension $\bar{\sigma}$ of σ on a final F^A -coalgebra. The construction relies on adding a family of auxiliary “buffer” operations to the syntax. These buffer operations depend only on A , and so the construction is uniform for all σ and F .

1. Introduction

In coalgebra, state-based systems are modelled as F -coalgebras where F is a functor that determines the system type. By varying F , we obtain (non)deterministic automata, (labelled) transition systems and many others, see (Rutten 2000) for an introduction and plenty of examples. Of particular importance are *final* coalgebras, which represent abstract behaviours of F -coalgebras. For this reason we refer to elements of a final F -coalgebra as F -behaviours. Examples of F -behaviours include streams, infinite trees, causal stream functions and processes. Thanks to the properties of final coalgebras, operations on them can be conveniently defined by *coinduction*.

In this paper, we focus on pointwise extensions of operations on F -behaviours to operations on F^A -behaviours, where F is an arbitrary functor, and A is a fixed set. Intuitively, F^A -coalgebras behave as F -coalgebras, but are additionally dependent on an external source of input from the alphabet A . For example, if $F X = B \times X$ then F^A -coalgebras are

[†] Supported by EPSRC grant EP/F042337/1.

Mealy machines with input in A and output in B , and F^A -behaviours are causal stream functions $f: A^\omega \rightarrow B^\omega$. We show that, in general, elements of the set \bar{Z} of F^A -behaviours can be thought of as certain functions from A^ω to a final F -coalgebra Z . An operation σ on Z can therefore be pointwise extended in the standard way to an operation $\bar{\sigma}$ on the function space Z^{A^ω} . Depending on σ , the operation $\bar{\sigma}$ restricts or not to an operation on \bar{Z} .

A well-structured way of defining operations on final coalgebras is by means of distributive laws of syntax over behaviour (cf. Turi and Plotkin 1997; Bartels 2004; Klin 2009), where syntax is given by an algebraic signature Σ and behaviour is given by a functor F . The rather abstract notion of a distributive law can often be formulated in a more intuitive fashion as a set of equations or rules. Concrete examples include Rutten’s behavioural differential equations (see e.g. Rutten 2003) and rules of structural operational semantics of processes (cf. Aceto et al. 2001).

Distributive laws provide a setting in which specification formats for varying syntax and behaviour types can be treated in a uniform way (using parametricity in Σ and F). In particular, the GSOS format of processes (cf. Aceto et al. 2001) generalises to an abstract GSOS format for arbitrary Σ and F . A further benefit of working in this more abstract setting is that a distributive law of Σ over F not only defines Σ -operations on F -behaviours, but also provides an inductive definition of an F -coalgebra on Σ -terms, and relates the final semantics of the latter with the initial semantics of the former.

Viewing specification formats as distributive laws, it makes sense to ask the following: given an operation σ , defined on a final F -coalgebra by a distributive law λ , is its pointwise extension $\bar{\sigma}$ an operation on a final F^A -coalgebra defined by another distributive law $\bar{\lambda}$, and can we obtain $\bar{\lambda}$ from λ in a systematic manner? The main technical contribution of this paper is an affirmative answer to that question under certain conditions. In Theorem 6.1, we show that definitions in a simple format can be extended in a straightforward manner. Then, in Theorem 6.3, we deal with the more expressive, abstract GSOS format, where the situation is more subtle. We show that GSOS-defined operations can also be extended pointwise, but the extension relies on a family of auxiliary operations that intuitively work as “input buffers” for F^A -coalgebras. It is worth noting that the choice of auxiliary operators depends only on the set A , but does not depend on λ , nor even on the behaviour functor F . In order to illustrate these general results, we provide several detailed examples. We mention that an extended abstract of this paper has appeared as (Hansen and Klin 2010).

The structure of the paper is as follows. In Section 2, basic facts on **Set**-functors, algebras and coalgebras are recalled, together with a few examples of F -coalgebras that are used in the paper. Section 3 contains two simple examples of operations on streams, and their pointwise extensions to Mealy machines. The techniques used in these examples motivate, and hopefully provide some intuition about, the abstract development of the subsequent sections. In Section 4 we show how to interpret elements of F^A -coalgebras as functions from A -streams to final F -coalgebras via their pointwise behaviour; based on this, we provide an abstract definition of pointwise extension of operations, illustrated by a few examples. In Section 5 we recall the distributive law approach to defining operations, as developed in (Turi and Plotkin 1997; Lenisa et al. 2004). Section 6 contains the two

main technical theorems as described above, followed by a few example applications in Section 7, and a brief discussion of directions for future work in Section 8. Proofs of the main theorems are put in an Appendix.

2. Preliminaries

In this section we fix some notation and provide the basic definitions of algebras and coalgebras needed in this paper. We assume that the reader is familiar with the notions of category, functor, natural transformation; the notions of adjunction and monad are mentioned a few times, but their detailed understanding is not necessary to follow the main development. All functors considered in this paper are endofunctors on **Set**, the category of sets and functions.

2.1. Strength and costrength

We recall some basic properties of **Set** functors that will be useful in the following.

For any set A , there is an adjunction $A \times - \dashv (-)^A$. The obvious unit and the counit of this adjunction will be denoted

$$\eta_X : X \rightarrow (A \times X)^A \quad \epsilon_X : A \times X^A \rightarrow X.$$

Any endofunctor F on **Set** has a *strength*, i.e., a map

$$\text{st}_{A,X}^F : A \times FX \rightarrow F(A \times X)$$

natural in A and X , defined by

$$\text{st}_{A,X}^F(a, t) = (F\eta^a)(t)$$

where $\eta^a : X \rightarrow A \times X$ is given by $\eta^a(x) = (a, x)$. Dually, every F has a *costrength*

$$\text{cs}_{A,X}^F : F(X^A) \rightarrow (FX)^A$$

natural in A and X , defined by

$$\text{cs}_{A,X}^F(t)(a) = (F\epsilon^a)(t)$$

where $\epsilon^a : X^A \rightarrow X$ is given by $\epsilon^a(f) = f(a)$.

2.2. Syntax via algebras

An *algebraic signature* Σ consists of a collection of function symbols $\{\sigma_i \mid i \in I\}$ where each σ_i has an arity $n_i \in \mathbb{N}$, $i \in I$. A Σ -*algebra* with carrier set X is a map $\prod_{i \in I} X^{n_i} \rightarrow X$ and we therefore identify a signature Σ with the functor $\Sigma X = \prod_{i \in I} X^{n_i}$. In general, given a functor G , a G -*algebra* is a pair $\langle X, \sigma \rangle$ where X is the carrier set and $\sigma : GX \rightarrow X$ is a function.

The set of Σ -*terms over a set (of variables)* X is denoted by $T_\Sigma X$; we shall omit the subscript in the following, as it will never lead to any confusion. In fact, T is a functor, and together with obvious natural transformations $\eta^T : Id \Longrightarrow T$ (interpretation of variables

as terms) and $\mu^T: TT \Longrightarrow T$ (glueing terms built of terms) it forms the so-called free monad over Σ . By structural induction on terms, any algebra $\sigma: \Sigma X \rightarrow X$ induces a function $\sigma^\sharp: TX \rightarrow X$ (i.e. term interpretation in σ). The T -algebra σ^\sharp satisfies axioms:

$$\sigma^\sharp \circ \eta_X^T = \text{id}_X \quad \sigma^\sharp \circ T\sigma^\sharp = \sigma^\sharp \circ \mu_X^T,$$

i.e., σ^\sharp is an *Eilenberg-Moore algebra* for the monad T . The construction of σ^\sharp from σ provides a 1-1 correspondence between Σ -algebras and Eilenberg-Moore T -algebras.

2.3. Behaviour via coalgebras

Coalgebra provides a uniform framework for studying the behaviour of systems such as automata and labelled transition systems. We only provide the basic definitions here. For a more thorough introduction to the theory of coalgebra we refer to (Rutten 2000). Formally, given a functor F , an F -coalgebra is a pair $\langle X, \xi \rangle$ where X is a set (called the carrier, or the set of states) and $\xi: X \rightarrow FX$ is a function (called the structure). Since $\xi: X \rightarrow FX$ implicitly defines the carrier X , we often refer to an F -coalgebra simply by its structure. Different types of systems emerge by varying F . We provide concrete examples in Section 2.4.

Behaviour preserving maps are formalised as the general notion of morphism between F -coalgebras. An F -coalgebra morphism $h: \langle X_1, \xi_1 \rangle \rightarrow \langle X_2, \xi_2 \rangle$ is a function $h: X_1 \rightarrow X_2$ such that $\xi_2 \circ h = Fh \circ \xi_1$. F -coalgebras and F -coalgebra morphisms form a category $\text{Coalg}(F)$. An abstract notion of behaviour is obtained via finality. An F -coalgebra $\langle Z, \zeta \rangle$ is *final*, if for every F -coalgebra $\langle X, \xi \rangle$ there is a unique F -coalgebra morphism $\text{beh}_{\langle X, \xi \rangle}$ (called the *final map*) from $\langle X, \xi \rangle$ to $\langle Z, \zeta \rangle$. A final F -coalgebra $\langle Z, \zeta \rangle$ can be seen as a system of behaviours, and we refer to the elements of Z as F -behaviours. By the so-called Lambek lemma (Lambek 1968), the structure ζ of a final coalgebra is always an isomorphism. A final F -coalgebra need not always exist due to cardinality reasons, (cf. Aczel and Mendler 1989), but all functors considered in this paper admit final coalgebras.

The existence and uniqueness of the final map give rise to a definition principle usually referred to as *coinduction*. We will use coinduction to define operations on the carriers of final coalgebras. Let F be a functor and $\langle Z, \zeta \rangle$ the final F -coalgebra. By defining an F -coalgebra structure $\xi: Z \times Z \rightarrow F(Z \times Z)$ the final map $\text{beh}_\xi: Z \times Z \rightarrow Z$ defines a binary operation \star on Z by coinduction. So the F -coalgebra structure ξ essentially specifies how $x \star y$ behaves for all $x, y \in Z$. More generally, for an algebraic signature Σ , an F -coalgebra structure on ΣZ induces a Σ -algebra on Z by coinduction as illustrated here:

$$\begin{array}{ccc} \Sigma Z & \xrightarrow{\sigma} & Z \\ \xi \downarrow & & \downarrow \zeta \\ F\Sigma Z & \xrightarrow{F\sigma} & FZ \end{array} \quad (1)$$

In Section 5 we will see how certain natural transformations correspond to various kinds of the specification ξ .

2.4. Examples

We shall now see a number of concrete examples of functors and their final coalgebras. All of these are well known, but we include them in some detail as they will be used later on for illustrating pointwise extensions.

Example 2.1 (Stream automata) Given a set A , a *stream automaton* (with output in A) is a coalgebra for the functor $A \times -$, i.e., it is a function $\langle o, d \rangle: X \rightarrow A \times X$ which maps an $x \in X$ to an output value $o(x) \in A$ and a next state $d(x) \in X$. We will use the notation $x \xrightarrow{a} y$ to denote that $o(x) = a$ and $d(x) = y$. The final $(A \times -)$ -coalgebra is obtained as the set A^ω of streams over A together with the head and tail maps, $\langle hd, tl \rangle: A^\omega \rightarrow A \times A^\omega$. The $(A \times -)$ -behaviour of a state x is the stream of outputs generated on transitions starting in x . We will use the following standard notation. A stream $\alpha \in A^\omega$ may be written $(\alpha(0), \alpha(1), \alpha(2), \dots)$ or as an ω -regular expression over A . For $\alpha \in A^\omega$ and $n \in \mathbb{N}$, $\alpha \upharpoonright_n = (\alpha(0), \dots, \alpha(n))$ is the prefix of α of length $n + 1$. For $\alpha \in A^\omega$ and $a \in A$, we denote with $a:\alpha$ the stream $(a, \alpha(0), \alpha(1), \dots)$. \triangleleft

Example 2.2 (Mealy machines) Given sets A and B , a $(B \times -)^A$ -coalgebra $m: X \rightarrow (B \times X)^A$ is a *Mealy machine* with input in A and output in B : for each state $x \in X$, $m(x) = \langle o_x, d_x \rangle: A \rightarrow B \times X$ defines for each $a \in A$, the output $o_x(a)$ and the next state $d_x(a)$. We write $x \xrightarrow{a|b} y$ if $o_x(a) = b$ and $d_x(a) = y$. The behaviour of a state x is the input-output mapping computed by $m: X \rightarrow (B \times X)^A$ when starting in x . Formally, $beh(x): A^\omega \rightarrow B^\omega$ maps $\alpha \in A^\omega$ to the stream $beh(x)(\alpha) \in B^\omega$ where for all $n \in \mathbb{N}$:

$$beh(x)(\alpha)(n) = o_{x_n}(\alpha(n)) \quad x_0 := x, \quad x_{n+1} := d_{x_n}(\alpha(n)). \quad (2)$$

From (2) it is clear that the n -th element of $beh(x)(\alpha)$ depends only on $\alpha(0), \dots, \alpha(n)$. A stream function $f: A^\omega \rightarrow B^\omega$ is *causal* if for all $n \in \mathbb{N}$, and all $\alpha, \beta \in A^\omega$:

$$\alpha \upharpoonright_n = \beta \upharpoonright_n \quad \implies \quad f(\alpha)(n) = f(\beta)(n).$$

The set $\Gamma = \{f: A^\omega \rightarrow B^\omega \mid f \text{ is causal}\}$ carries a final Mealy structure as shown in (Rutten 2006). We briefly summarise the construction. Let $f \in \Gamma$ and $a \in A$. We write $f(a: -)$ for the stream function which maps $\alpha \in A^\omega$ to $f(a:\alpha) \in B^\omega$. The *initial output* $f[a]$ and the *stream function derivative* f_a of f on a are defined as:

$$\begin{aligned} f[a] &:= hd \circ f(a:-) \in A^\omega \rightarrow B \\ f_a &:= tl \circ f(a:-) \in A^\omega \rightarrow B^\omega \end{aligned} \quad (3)$$

Since f is causal, $f[a]$ is constant (so we can consider $f[a]$ an element of B) and f_a is causal, hence by defining for all $f \in \Gamma$ and $a \in A$,

$$\gamma(f)(a) = \langle f[a], f_a \rangle,$$

γ is a map of type $\gamma: \Gamma \rightarrow (B \times \Gamma)^A$, and it can be shown that $\langle \Gamma, \gamma \rangle$ is a final Mealy machine. \triangleleft

Example 2.3 (Partial maps) We denote the one-element set by $1 = \{\perp\}$. If we let \perp represent the undefined value, then a $(1 + -)$ -coalgebra $\xi: X \rightarrow 1 + X$ can be seen as

a partial function on X , and we write $x \longrightarrow y$ if $\xi(x) = y$ (for $y \in 1 + X$). The final $(1 + -)$ -coalgebra consists of the natural numbers extended with ω together with the predecessor function $\text{pred} : \mathbb{N} + \{\omega\} \rightarrow 1 + \mathbb{N} + \{\omega\}$ where $\text{pred}(0) = \perp$, $\text{pred}(\omega) = \omega$ and $\text{pred}(n) = n - 1$ for all $n \in \mathbb{N}_{>0}$. The behaviour of a state x is the number of steps that can be made from x . \triangleleft

Example 2.4 (Partial DLTSs) A $(1 + -)^A$ -coalgebra $t : X \rightarrow (1 + X)^A$ can be seen as a partial, deterministic labelled transition system where the a -successor of a state x is given by $t(x)(a)$ in case $t(x)(a) \neq \perp$. We write $x \xrightarrow{a} y$ if $t(x)(a) = y$ (for $y \in 1 + X$).

The final $(1 + -)^A$ -coalgebra $\langle \mathcal{L}, D \rangle$ (cf. Rutten 1991) has as its carrier the set of non-empty, prefix-closed languages over A , i.e.,

$$\mathcal{L} = \{L \subseteq A^* \mid L \neq \emptyset; \text{ for all } u, v \in A^* : uv \in L \Rightarrow u \in L\}.$$

For $L \in \mathcal{L}$, the function $D(L) : A \rightarrow 1 + \mathcal{L}$ is defined by

$$D(L)(a) = \begin{cases} L_a := \{w \in A^* \mid aw \in L\} & \text{if } a \in L, \\ \perp & \text{if } a \notin L. \end{cases}$$

L_a is called the language derivative of L with respect to a . Note that if $a \in L$ then L_a is non-empty, and if L is prefix-closed then L_a is prefix-closed, hence D is well defined. The behaviour of a state x is the set $L(x)$ of finite words that label some path in t starting in x . We illustrate with a small example:

$$\begin{array}{ccc} \begin{array}{c} b \curvearrowright \\ x_0 \xrightarrow{a} x_1 \xrightarrow{a} x_2 \xrightarrow{a} x_3 \\ \curvearrowleft b \end{array} & & \begin{array}{l} L(x_0) = b^* + b^*a(ab)^* + b^*a(ab)^*a + b^*a(ab)^*aa \\ L(x_1) = (ab)^* + (ab)^*a + (ab)^*aa \\ L(x_2) = (ba)^* + (ba)^*a + (ba)^*b \\ L(x_3) = \{\varepsilon\} \end{array} \end{array}$$

Example 2.5 (Nondeterministic systems) We denote by \mathcal{P}_ω the finitary (covariant) powerset functor which maps a set X to the set of all its finite subsets. A \mathcal{P}_ω -coalgebra $\xi : X \rightarrow \mathcal{P}_\omega X$ models a *finitely branching (nondeterministic) transition system*, and we write $x \longrightarrow y$ if $y \in \xi(x)$, and $x \not\longrightarrow$ if $\xi(x) = \emptyset$.

The final \mathcal{P}_ω -coalgebra $\langle T, \tau \rangle$ is usually described as the set of strongly extensional, finitely branching trees together with the subtree relation (cf. Worrell 2005). An alternative way of thinking about the \mathcal{P}_ω -behaviour of a state x is as the bisimilarity quotient of the subsystem generated from x (cf. Rutten and Turi 1994). \triangleleft

Example 2.6 (LTSSs) A $(\mathcal{P}_\omega -)^A$ -coalgebra $t : X \rightarrow (\mathcal{P}_\omega X)^A$ is an image-finite labelled transition system with state space X . We use the notation $x \xrightarrow{a} y$ if $y \in t(x)(a)$, and $x \not\xrightarrow{a}$ if $t(x)(a) = \emptyset$, for all $x, y \in X$ and $a \in A$. The final $(\mathcal{P}_\omega -)^A$ -coalgebra $\langle P, \phi \rangle$ is carried by the set P of image-finite *processes*, which can also be described as the $(\mathcal{P}_\omega -)^A$ -bisimulation classes of all finitely branching, A -labelled trees with possibly infinite branches (cf. Rutten and Turi 1994). The behaviour of a state x is thus the $(\mathcal{P}_\omega -)^A$ -bisimulation quotient of the substructure generated from x . \triangleleft

3. Motivating Examples

To motivate the abstract development of Sections 4 and 6, and provide some intuitions about the techniques used there, we shall begin with two concrete examples of operations on the set \mathbb{R}^ω of streams over the real numbers, i.e. the carrier of a final $(\mathbb{R} \times -)$ -coalgebra, and show how to extend their definitions to operations on causal stream functions.

As explained in (Rutten 2003), operations on streams can be defined with the help of *stream differential equations*, where the resulting stream is unambiguously specified by its initial value (head) and its stream derivative (tail) in terms of the initial values and the derivatives of the argument streams.

Example 3.1 The element-wise addition \oplus of streams over \mathbb{R} is defined by the stream differential equation:

$$hd(x_0 \oplus y_0) = hd(x_0) + hd(y_0) \quad tl(x_0 \oplus y_0) = tl(x_0) \oplus tl(y_0)$$

The above stream differential equation can be formulated as a family of derivation rules:

$$\frac{x_0 \xrightarrow{a} x_1 \quad y_0 \xrightarrow{b} y_1}{x_0 \oplus y_0 \xrightarrow{a+b} x_1 \oplus y_1}$$

where a, b range over \mathbb{R} . ◁

The operation $\oplus : \mathbb{R}^\omega \times \mathbb{R}^\omega \rightarrow \mathbb{R}^\omega$ adds two streams of numbers element-wise:

$$\alpha \oplus \beta = (\alpha(0) + \beta(0), \alpha(1) + \beta(1), \alpha(2) + \beta(2), \dots)$$

for all $\alpha, \beta \in \mathbb{R}^\omega$. The details of how this follows from the equations (or, equivalently, the rules) can be found in (Rutten 2003). An alternative but equivalent explanation will be provided in Section 5, where the equations/rules are interpreted abstractly as distributive laws.

Here is another, deceptively similar example:

Example 3.2 The operation $\boxplus : \mathbb{R}^\omega \times \mathbb{R}^\omega \rightarrow \mathbb{R}^\omega$ is defined by:

$$\begin{aligned} hd(x_0 \boxplus y_0) &= hd(x_0) + hd(y_0) & \text{i.e.} & \quad \frac{x_0 \xrightarrow{a} x_1 \quad y_0 \xrightarrow{b} y_1}{x_0 \boxplus y_0 \xrightarrow{a+b} x_0 \boxplus y_1} \\ tl(x_0 \boxplus y_0) &= x_0 \boxplus tl(y_0) \end{aligned} \quad (4)$$

The operation \boxplus adds the head element of its first argument to each of the elements of the second argument stream:

$$\alpha \boxplus \beta = (\alpha(0) + \beta(0), \alpha(0) + \beta(1), \alpha(0) + \beta(2), \dots)$$

for all $\alpha, \beta \in \mathbb{R}^\omega$. ◁

The operations \oplus and \boxplus can be extended pointwise to the set of causal stream functions $\Gamma_{\mathbb{R}} = \{f : \mathbb{R}^\omega \rightarrow \mathbb{R}^\omega \mid f \text{ is causal}\}$ by defining

$$\begin{aligned} (f \bar{\oplus} g)(\alpha) &= f(\alpha) \oplus g(\alpha) \\ (f \bar{\boxplus} g)(\alpha) &= f(\alpha) \boxplus g(\alpha) \end{aligned}$$

for all $f, g \in \Gamma_{\mathbb{R}}$ and $\alpha \in \mathbb{R}^\omega$. Note in particular, that $f \bar{\oplus} g$ and $f \bar{\boxplus} g$ are again causal, so $\bar{\oplus}$ and $\bar{\boxplus}$ are well defined as operations on $\Gamma_{\mathbb{R}}$. In general, given an n -ary operation σ on a set Y , the *pointwise extension* of σ to the function space Y^X , is the operation $\bar{\sigma}: (Y^X)^n \rightarrow Y^X$ defined for all $f_0, \dots, f_{n-1}: X \rightarrow Y$ by

$$\bar{\sigma}(f_0, \dots, f_{n-1})(x) = \sigma(f_0(x), \dots, f_{n-1}(x)).$$

The question is, can we give a rule-based definition of the pointwise extensions $\bar{\oplus}$ and $\bar{\boxplus}$, possibly based on the defining rules for \oplus and \boxplus , respectively?

Recall from Example 2.2 that $\Gamma_{\mathbb{R}}$ is the final $(\mathbb{R} \times -)^{\mathbb{R}}$ -coalgebra under the operations initial output $f[a]$ and stream function derivative f_a . Using the notation $f \xrightarrow{a|f[a]} f_a$, the pointwise extension $\bar{\oplus}: \Gamma_{\mathbb{R}} \times \Gamma_{\mathbb{R}} \rightarrow \Gamma_{\mathbb{R}}$ can be defined as follows:

$$\frac{x_0 \xrightarrow{a|b} x_1 \quad y_0 \xrightarrow{a|c} y_1}{x_0 \bar{\oplus} y_0 \xrightarrow{a|b+c} x_1 \bar{\oplus} y_1} \quad (5)$$

To see how this rule works, suppose that $\alpha = (a_0, a_1, a_2, \dots) \in \mathbb{R}^\omega$ and $f, g \in \Gamma_{\mathbb{R}}$ with $f(\alpha) = (b_0, b_1, b_2, \dots)$ and $g(\alpha) = (c_0, c_1, c_2, \dots)$. It follows, in particular, that for the repeated derivatives we have: $f_{a_0 \dots a_n}[a_{n+1}] = b_{n+1}$ and $g_{a_0 \dots a_n}[a_{n+1}] = c_{n+1}$ for all $n \in \mathbb{N}$. The rule in (5) induces the following transitions of $f \bar{\oplus} g$ on input α :

$$f \bar{\oplus} g \xrightarrow{a_0|b_0+c_0} f_{a_0} \bar{\oplus} g_{a_0} \xrightarrow{a_1|b_1+c_1} f_{a_0 a_1} \bar{\oplus} g_{a_0 a_1} \xrightarrow{a_2|b_2+c_2} \dots$$

and we see that $(f \bar{\oplus} g)(\alpha) = f(\alpha) \oplus g(\alpha)$. Since α was arbitrary, $\bar{\oplus}$ is indeed the pointwise extension of \oplus .

Let us now consider the pointwise extension of \boxplus . By analogy with (5), we could try to define $\bar{\boxplus}: \Gamma_{\mathbb{R}} \times \Gamma_{\mathbb{R}} \rightarrow \Gamma_{\mathbb{R}}$ with the following rule:

$$\frac{x_0 \xrightarrow{a|b} x_1 \quad y_0 \xrightarrow{a|c} y_1}{x_0 \bar{\boxplus} y_0 \xrightarrow{a|b+c} x_0 \bar{\boxplus} y_1} \quad (6)$$

For $f, g \in \Gamma_{\mathbb{R}}$, the rule (6) implies that for all $a \in \mathbb{R}$:

$$\begin{aligned} (f \bar{\boxplus} g)[a] &= f[a] + g[a] \\ (f \bar{\boxplus} g)_a &= f \bar{\boxplus} g_a \end{aligned} \quad (7)$$

However, the induced operation on $\Gamma_{\mathbb{R}}$ is *not* the pointwise extension of \boxplus . To see this, consider $\text{id} \bar{\boxplus} \text{id}$ where id is the identity on \mathbb{R}^ω . Note that in particular, $\text{id}[a] = a$ and $\text{id}_a = \text{id}$ for all $a \in \mathbb{R}$. The rule (6) then yields the following transitions on input $\alpha = (a_0, a_1, a_2, \dots) \in \mathbb{R}^\omega$:

$$\text{id} \bar{\boxplus} \text{id} \xrightarrow{a_0|a_0+a_0} \text{id} \bar{\boxplus} \text{id} \xrightarrow{a_1|a_1+a_1} \text{id} \bar{\boxplus} \text{id} \xrightarrow{a_2|a_2+a_2} \dots$$

hence $(\text{id} \bar{\boxplus} \text{id})(\alpha) \neq \alpha \boxplus \alpha = (a_0 + a_0, a_0 + a_1, a_0 + a_2, \dots)$.

To understand what goes wrong, we must look into the structure of the final Mealy machine. Recall from Example 2.2 that for all $f \in \Gamma_{\mathbb{R}}$ and $a \in \mathbb{R}$, $f_a = \text{tl}(f(a: -))$. In order for $\bar{\boxplus}$ to be the pointwise extension of \boxplus , the following must hold for all $f, g \in \Gamma_{\mathbb{R}}$, $a \in \mathbb{R}$ and $\alpha \in \mathbb{R}^\omega$:

$$(f \bar{\boxplus} g)(a: \alpha) = f(a: \alpha) \boxplus g(a: \alpha).$$

By taking tails on both sides, and applying the rule (4) for \boxplus we get

$$(f\bar{\boxplus}g)_a(\alpha) = f(a:\alpha) \boxplus g_a(\alpha). \quad (8)$$

Comparing (8) with (7), we see that in the definition of $(f\bar{\boxplus}g)_a$, f should be replaced with the function $f(a:-)$. In order to give a rule-based definition of $\bar{\boxplus}$ (which applies not only to $\Gamma_{\mathbb{R}}$, but to all Mealy machines), it makes sense to use a family of auxiliary, unary operations $\{a \triangleright - \mid a \in \mathbb{R}\}$. These operations act like one-element input buffers: for a Mealy machine $m: X \rightarrow (\mathbb{R} \times X)^{\mathbb{R}}$ and $x \in X$, $a \triangleright x$ behaves like x as if it had seen an a already. Formally, $beh(a \triangleright x)(\alpha) = beh(x)(a:\alpha)$ for all $\alpha \in \mathbb{R}^{\omega}$ and $a \in \mathbb{R}$. We can now define $\bar{\boxplus}$, together with the buffer operations, by rules:

$$\frac{x_0 \xrightarrow{a|b} x_1 \quad y_0 \xrightarrow{a|c} y_1}{x_0 \bar{\boxplus} y_0 \xrightarrow{a|b+c} (a \triangleright x_0) \bar{\boxplus} y_1} \quad \frac{x_0 \xrightarrow{a|b} x_1}{a \triangleright x_0 \xrightarrow{c|b} c \triangleright x_1} \quad (9)$$

To illustrate how this definition works, let $\alpha = (a_0, a_1, a_2, \dots) \in \mathbb{R}^{\omega}$ and $f, g \in \Gamma_{\mathbb{R}}$ with $f(\alpha) = (b_0, b_1, b_2, \dots)$ and $g(\alpha) = (c_0, c_1, c_2, \dots)$. First, the rule for $a \triangleright -$ gives us,

$$\begin{aligned} f \xrightarrow{a_0|b_0} f_{a_0} &\implies (a_0 \triangleright f) \xrightarrow{a_1|b_0} (a_1 \triangleright f_{a_0}) \\ &\implies (a_1 \triangleright a_0 \triangleright f) \xrightarrow{a_2|b_0} (a_2 \triangleright a_1 \triangleright f_{a_0}) \\ &\implies (a_2 \triangleright a_1 \triangleright a_0 \triangleright f) \xrightarrow{a_3|b_0} (a_3 \triangleright a_2 \triangleright a_1 \triangleright f_{a_0}) \\ &\implies \dots \end{aligned}$$

We now find that,

$$\begin{aligned} f\bar{\boxplus}g &\xrightarrow{a_0|b_0+c_0} (a_0 \triangleright f)\bar{\boxplus}g_{a_0} \\ &\xrightarrow{a_1|b_0+c_1} (a_1 \triangleright a_0 \triangleright f)\bar{\boxplus}g_{a_0 a_1} \\ &\xrightarrow{a_2|b_0+c_2} (a_2 \triangleright a_1 \triangleright a_0 \triangleright f)\bar{\boxplus}g_{a_0 a_1 a_2} \\ &\xrightarrow{a_3|b_0+c_3} \dots \end{aligned}$$

So $\bar{\boxplus}$ indeed behaves as the pointwise extension of \boxplus .

In the rest of this paper, we show that the constructions used in the above examples apply more generally. In particular, the idea of defining pointwise extensions by adding input buffer operations works not only for the specific stream operation \boxplus , but in fact for all GSOS-defined operations on F -behaviours, for any functor F . The meaning of the phrase ‘‘GSOS-defined’’ is explained in detail in Section 5. Before that, however, we need to understand the notion of pointwise extension for operations on final F -coalgebras.

4. Pointwise Behaviour and Pointwise Extensions of Operations

In this section, we introduce the notion of pointwise behaviour of F -coalgebras with input. From now on, let A be a fixed set. An F -coalgebra with input (in A) is a coalgebra for the functor composition $F^A = (-)^A \circ F$. That is, $F^A(X) = (FX)^A$ and for a function $f: X \rightarrow Y$, $F^A(f): (FX)^A \rightarrow (FY)^A$ is defined by $F^A(f)(t)(a) = (Ff)(t(a))$. Our motivating examples in Section 3 focused on the case where $F = (B \times -)$, that is, F^A -coalgebras are Mealy machines, and F^A -behaviours are causal stream functions of type

$f: A^\omega \rightarrow B^\omega$. Evaluating such a causal stream function at an input stream $\alpha \in A^\omega$ yields an element in B^ω , the set of F -behaviours. We will show that for any functor F , under the assumption that the final coalgebras for F and F^A exist, we can evaluate F^A -behaviours at input streams from A^ω to obtain F -behaviours. This observation relies on a construction which is similar to the wreath product of automata (cf. Carton 2000; Straubing 1989).

Definition 4.1 Let $s: Y \rightarrow A \times Y$ be an $(A \times -)$ -coalgebra. For an F^A -coalgebra $m: X \rightarrow (FX)^A$, define the F -coalgebra

$$s \ltimes m: Y \times X \rightarrow F(Y \times X)$$

to be the function composition:

$$\begin{aligned} Y \times X &\xrightarrow{s \ltimes m} (A \times Y) \times (FX)^A \xrightarrow{\cong} Y \times (A \times (FX)^A) \\ &\xrightarrow{\text{id}_Y \times \epsilon_{FX}} Y \times FX \xrightarrow{\text{st}_{Y,X}^F} F(Y \times X) \end{aligned}$$

◀

If $\langle Z, \zeta \rangle$ is a final F -coalgebra, then for all $s: Y \rightarrow A \times Y$ and $m: X \rightarrow (FX)^A$, we get by finality a unique F -coalgebra morphism $\text{beh}_{s \ltimes m}: Y \times X \rightarrow Z$:

$$\begin{array}{ccc} Y \times X & \xrightarrow{\text{beh}_{s \ltimes m}} & Z \\ s \ltimes m \downarrow & & \downarrow \zeta \\ F(Y \times X) & \xrightarrow{F \text{beh}_{s \ltimes m}} & FZ \end{array}$$

We call $\text{beh}_{s \ltimes m}(\langle y, x \rangle)$ the *pointwise F -behaviour of x at y* for $\langle y, x \rangle \in Y \times X$. In this paper we are mainly interested in the case where s is the final $(A \times -)$ -coalgebra $\langle A^\omega, \langle hd, tl \rangle \rangle$, and m is the final F^A -coalgebra $\langle \bar{Z}, \bar{\zeta} \rangle$, assuming it exists. The pointwise F -behaviour map can then be thought of as evaluating F^A -behaviours at input streams. We indicate this by using the notation ev instead of $\text{beh}_{\langle hd, tl \rangle \ltimes \bar{\zeta}}$:

$$\begin{array}{ccc} A^\omega \times \bar{Z} & \xrightarrow{\text{ev}} & Z \\ \langle hd, tl \rangle \ltimes \bar{\zeta} \downarrow & & \downarrow \zeta \\ F(A^\omega \times \bar{Z}) & \xrightarrow{F \text{ev}} & FZ \end{array} \quad (10)$$

Using the map ev , elements of \bar{Z} can be interpreted as functions from A^ω to Z . In general, not all such functions arise from elements of \bar{Z} ; nevertheless, this allows us to meaningfully speak of some operations (algebras) on \bar{Z} as pointwise extensions of operations (algebras) on Z . Formally:

Definition 4.2 For a signature Σ , an algebra $\bar{\sigma}: \Sigma \bar{Z} \rightarrow \bar{Z}$ is a *pointwise extension* of

$\sigma : \Sigma Z \rightarrow Z$ if the diagram:

$$\begin{array}{ccc}
 A^\omega \times \Sigma \bar{Z} & \xrightarrow{\text{st}_{A^\omega, \bar{Z}}^\Sigma} & \Sigma(A^\omega \times \bar{Z}) \xrightarrow{\Sigma \text{ev}} \Sigma Z \\
 \text{id} \times \bar{\sigma} \downarrow & & \downarrow \sigma \\
 A^\omega \times \bar{Z} & \xrightarrow{\text{ev}} & Z
 \end{array} \tag{11}$$

commutes. \blacktriangleleft

We now show a few examples of pointwise behaviour and pointwise extensions.

Example 4.3 In our main example of streams and Mealy machines, the relevant functors are $F = (B \times -)$ and $F^A = (B \times -)^A$, F -behaviours are streams over B and F^A -behaviours are causal stream functions of type $f: A^\omega \rightarrow B^\omega$ (cf. Example 2.2). The input evaluation and pointwise behaviour of causal stream functions are illustrated in the diagram below. To ease notation, we write α_0 and α' for $hd(\alpha)$ and $tl(\alpha)$, respectively.

$$\begin{array}{ccccc}
 A^\omega \times \Gamma & & \langle \alpha, f \rangle & \xrightarrow{\text{ev}} & f(\alpha) & & B^\omega \\
 \langle hd, tl \rangle \times \gamma \downarrow & & \downarrow & & \downarrow & & \downarrow \langle hd, tl \rangle \\
 B \times (A^\omega \times \Gamma) & & \langle f[\alpha_0], \langle \alpha', f_{\alpha_0} \rangle \rangle & \xrightarrow{\text{id} \times \text{ev}} & \langle f[\alpha_0], f_{\alpha_0}(\alpha') \rangle & & B \times B^\omega
 \end{array}$$

Clearly, for all $f \in \Gamma$ and $\alpha \in A^\omega$, $\text{ev}(\langle \alpha, f \rangle) = f(\alpha)$.

In Section 3 we have already seen examples of pointwise extensions of stream operations to causal stream functions. We note, however, that not all stream operations can be extended pointwise to Γ . A simple example is given by the tail operation $tl: B^\omega \rightarrow B^\omega$. Its pointwise extension $\bar{tl}: \Gamma \rightarrow \Gamma$ would have to be defined for $f \in \Gamma$ and $\alpha \in A^\omega$ by $\bar{tl}(f)(\alpha) = tl(f(\alpha))$. But then $\bar{tl}(f)$ is not causal, in general. For example, taking $A = B$ and $f = \text{id}$, then $\bar{tl}(\text{id}) = tl$ which is not causal, so \bar{tl} is not well defined as an operation on Γ .

On the other hand, causal stream operations *can* be pointwise extended to Γ . More precisely, we say that a k -ary operation $\sigma: (B^\omega)^k \rightarrow B^\omega$ is causal if for all $\beta_0, \dots, \beta_{k-1} \in B^\omega$ and all $n \in \mathbb{N}$, $\sigma(\beta_0, \dots, \beta_{k-1})(n)$ depends only on $\beta_i \upharpoonright_n$ for $i \in \{0, \dots, k-1\}$. It is straightforward to show that if $\sigma: (B^\omega)^k \rightarrow B^\omega$ is causal, then its pointwise extension $\bar{\sigma}$ preserves causality, i.e., for $f_0, \dots, f_{k-1} \in \Gamma$, $\bar{\sigma}(f_0, \dots, f_{k-1}): A^\omega \rightarrow B^\omega$ is causal, and hence $\bar{\sigma}$ is a well defined operation on Γ . \triangleleft

Example 4.4 Consider the functor $FX = 1 + X$. Recall from Examples 2.3 and 2.4 that the set of F -behaviours is the extended natural numbers, and the final F^A -coalgebra $\langle \mathcal{L}, D \rangle$ is carried by the set of non-empty, prefix-closed languages. A transition in the F -coalgebra $\langle hd, tl \rangle \times D$ is defined by:

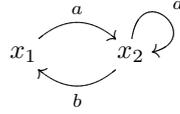
$$\begin{array}{ccc}
 A^\omega \times \mathcal{L} & \xrightarrow{\langle hd, tl \rangle \times D} & 1 + (A^\omega \times \mathcal{L}) \\
 \langle \alpha_0 : \alpha', L \rangle & \longmapsto & \begin{cases} \langle \alpha', L_{\alpha_0} \rangle & \text{if } \alpha_0 \in L \\ \perp & \text{if } \alpha_0 \notin L \end{cases}
 \end{array}$$

Given a state x in an F^A -coalgebra $t: X \rightarrow (1 + X)^A$ and an $\alpha \in A^\omega$, the pointwise F -behaviour of x on α is the number of steps that can be made in t starting from x on input α . More precisely,

$$\text{ev}(\langle \alpha, x \rangle) = \sup\{n \leq \omega \mid \alpha \upharpoonright_n \in L(x)\} \in \mathbb{N} + \{\omega\}.$$

In particular, if α labels an infinite path starting in x , then $\text{ev}(\langle \alpha, x \rangle) = \omega$.

As an example, consider the following (fragment of the final) $(1 + -)^A$ -coalgebra with input alphabet $A = \{a, b\}$:



We list a few examples of pointwise behaviours:

$$\begin{aligned} \text{ev}(\langle a^\omega, x_1 \rangle) &= \omega, & \text{ev}(\langle a^\omega, x_2 \rangle) &= \omega, \\ \text{ev}(\langle b^\omega, x_1 \rangle) &= 0, & \text{ev}(\langle b^\omega, x_2 \rangle) &= 1, \\ \text{ev}(\langle a^n b^\omega, x_1 \rangle) &= n + 1, & \text{ev}(\langle a^n b^\omega, x_2 \rangle) &= n + 1, \quad \forall n > 0. \end{aligned}$$

Natural operations on F -behaviours are max (\vee), plus ($+$) and times (\cdot) in the extended arithmetic where $\omega + x = \omega$ for all $x \in \{\omega\} + \mathbb{N}$ and $\omega \cdot x = \omega$ for all $x \in \{\omega\} + \mathbb{N}_{>0}$ and $\omega \cdot 0 = 0$. Using the above example, the pointwise extensions of these operations should satisfy:

$$\begin{aligned} \text{ev}(\langle a^\omega, x_1 \bar{\vee} x_2 \rangle) &= \omega, & \text{ev}(\langle b^\omega, x_1 \bar{\vee} x_2 \rangle) &= 1, \\ \text{ev}(\langle a^\omega, x_1 \bar{\cdot} x_2 \rangle) &= \omega, & \text{ev}(\langle b^\omega, x_1 \bar{\cdot} x_2 \rangle) &= 1, \\ \text{ev}(\langle a^\omega, x_1 \bar{+} x_2 \rangle) &= \omega, & \text{ev}(\langle b^\omega, x_1 \bar{+} x_2 \rangle) &= 0, \\ \\ \text{ev}(\langle a^n b^\omega, x_1 \bar{\vee} x_2 \rangle) &= n + 1, \\ \text{ev}(\langle a^n b^\omega, x_1 \bar{\cdot} x_2 \rangle) &= 2n + 2, \\ \text{ev}(\langle a^n b^\omega, x_1 \bar{+} x_2 \rangle) &= n^2 + 2n + 1, \quad \forall n > 0. \end{aligned}$$

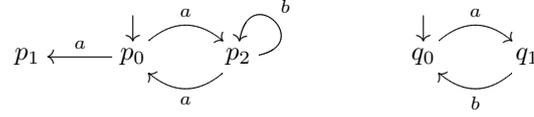
◁

Example 4.5 Recall from Examples 2.5 and 2.6 that the final \mathcal{P}_ω -coalgebra $\langle T, \tau \rangle$ consists of all strongly extensional, finitely branching trees with the subtree relation, and the final \mathcal{P}_ω^A -coalgebra $\langle P, \phi \rangle$ is carried by the set of image-finite processes. For $\alpha \in A^\omega$ and $p \in P$,

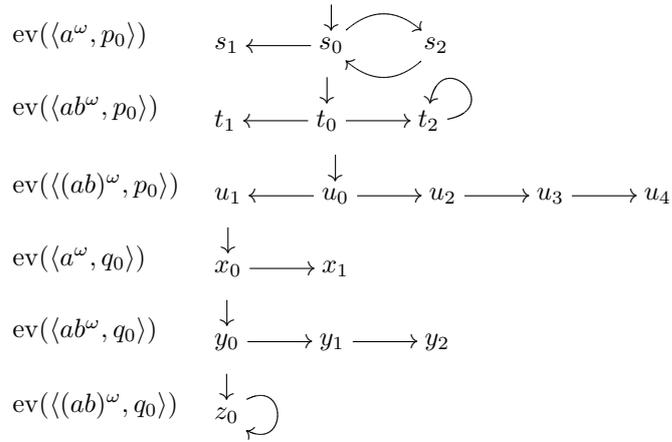
$$\begin{aligned} A^\omega \times P &\xrightarrow{\langle hd, tl \rangle \times \phi} \mathcal{P}_\omega(A^\omega \times P) \\ \langle \alpha, p \rangle &\longmapsto \{ \langle \alpha', q \rangle \mid p \xrightarrow{\alpha_0} q \}. \end{aligned}$$

Evaluating $p \in P$ at an input stream $\alpha \in A^\omega$ can be seen as removing from p the paths that are not labelled by a prefix of α , and dropping labels. The pointwise behaviour is obtained by quotienting the resulting tree with \mathcal{P}_ω -bisimilarity. For example, let $A =$

$\{a, b\}$ and let p_0 and q_0 be the processes:



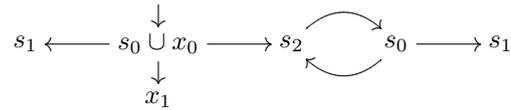
Then we find that the pointwise behaviours of p_0 and q_0 on a^ω , ab^ω and $(ab)^\omega$, respectively, are:



We consider the following operations on F -behaviours s and t . The *join* $s \cup t$ is a root node which has all the subtrees of s and t as subtrees. The *composition* $s; t$ plugs in t at all leaf nodes of s . The *interleaving* $s \otimes t$ is a tree version of shuffle product. These operations are defined by the following rules:

$$\begin{array}{l}
 \text{join :} \quad \frac{x_0 \longrightarrow x_1}{x_0 \cup y \longrightarrow x_1} \quad \frac{y_0 \longrightarrow y_1}{x \cup y_0 \longrightarrow y_1} \\
 \\
 \text{composition :} \quad \frac{x_0 \longrightarrow x_1}{x_0; y \longrightarrow x_1; y} \quad \frac{x \not\longrightarrow \quad y_0 \longrightarrow y_1}{x; y_0 \longrightarrow y_1} \\
 \\
 \text{interleaving :} \quad \frac{x_0 \longrightarrow x_1 \quad y_0 \longrightarrow y_1}{x_0 \otimes y_0 \longrightarrow (x_1 \otimes y_0) \cup (x_0 \otimes y_1)}
 \end{array}$$

We construct the join $s_0 \cup x_0$ by first computing transitions using the rules for \cup :



Quotienting with bisimilarity, we find that $s_0 \cup x_0 = s_0$. Compositions are also easily

In the above examples, pointwise extensions of operations were calculated on some specific argument values. To check whether these pointwise extensions are well-defined in general, and whether they can be defined in a rule-based fashion as in Section 3, it is necessary to understand those rule-based specifications in a general coalgebraic setting. This is the aim of the next section.

5. Distributive Laws and GSOS Rules

The equation- and rule-based definitions in the examples of Sections 3 and 4 are special cases of a general framework for defining operations on final coalgebras, parameterized both by the behaviour functor F and the signature Σ of operations. The framework, developed in (Turi and Plotkin 1997), is based on the abstract notion of distributive law.

Basic distributive laws. Let F be a behaviour functor, $\langle Z, \zeta \rangle$ the final F -coalgebra, and Σ an algebraic signature. A natural transformation $\lambda: \Sigma F \Rightarrow F\Sigma$, i.e. a distributive law of Σ over F , induces a Σ -algebra on Z by finality, as the unique F -coalgebra map from $\lambda_Z \circ \Sigma\zeta$ to $\langle Z, \zeta \rangle$. This is illustrated in the following diagram:

$$\begin{array}{ccc}
 \Sigma Z & \xrightarrow{\Sigma\zeta} & \Sigma FZ & \xrightarrow{\lambda_Z} & F\Sigma Z \\
 \sigma \downarrow & & & & \downarrow F\sigma \\
 Z & \xrightarrow{\zeta} & FZ & & FZ
 \end{array}
 \quad (12)$$

For instance, the definition of \oplus in Example 3.1 induces a natural transformation

$$\lambda: (\mathbb{R} \times -) \times (\mathbb{R} \times -) \Longrightarrow \mathbb{R} \times (- \times -)$$

(i.e. $\Sigma X = X \times X$ and $FX = \mathbb{R} \times X$) whose X -component is given by:

$$\begin{array}{ccc}
 \lambda_X: (\mathbb{R} \times X) \times (\mathbb{R} \times X) & \Longrightarrow & \mathbb{R} \times (X \times X) \\
 \langle \langle a, x_1 \rangle, \langle b, y_1 \rangle \rangle & \mapsto & \langle a + b, \langle x_1, y_1 \rangle \rangle.
 \end{array}
 \quad (13)$$

It is straightforward to check that the operation $\sigma: \mathbb{R}^\omega \times \mathbb{R}^\omega \rightarrow \mathbb{R}^\omega$ arising from this λ as in (12), is the expected operation \oplus induced by equations of Example 3.1 according to (Rutten 2003).

Monadic distributive laws. In specifications associated with basic distributive laws, all expressions on the right-hand sides of rule conclusions must be Σ -terms of depth exactly 1. Definitions of some useful operations do not conform to this restriction.

For an easy example, consider for $FX = \mathbb{R} \times X$ a unary ‘‘head replacement’’ operation $a/-: \mathbb{R}^\omega \rightarrow \mathbb{R}^\omega$, defined equivalently by equations or rules by:

$$\begin{array}{ccc}
 hd(a/x_0) & = & a \\
 tl(a/x_0) & = & tl(x_0)
 \end{array}
 \quad \text{i.e.} \quad
 \frac{x_0 \xrightarrow{b} x_1}{a/x_0 \xrightarrow{a} x_1}.$$

Note how the conclusion of the above rule is a variable, i.e., a term of depth 0 rather than 1. For this reason, the above definition does not correspond to a basic distributive

law $\lambda: \Sigma F \Longrightarrow F\Sigma$. There are also useful examples where the relevant terms have depth more than 1 (see e.g. the interleaving operation in Example 4.5).

To deal with such examples, one can consider natural transformations of the type $\rho: \Sigma F \Longrightarrow FT$; these allow arbitrary Σ -terms where only terms of depth 1 were allowed. Such natural transformations induce Σ -operations on final F -coalgebras much the same as basic distributive laws, as the unique maps $\sigma: \Sigma Z \rightarrow Z$ that make the diagram:

$$\begin{array}{ccccc}
 \Sigma Z & \xrightarrow{\Sigma\zeta} & \Sigma FZ & \xrightarrow{\rho z} & FTZ \\
 \sigma \downarrow \text{dotted} & & & & \downarrow F\sigma^\sharp \\
 Z & \xrightarrow[\cong]{\zeta} & & & FZ,
 \end{array} \tag{14}$$

commute, where σ^\sharp is the T -algebra corresponding to σ as described in Section 2.2.

An alternative way to understand this definition, and convince oneself that σ as above indeed exists uniquely, is to observe that transformations $\rho: \Sigma F \Longrightarrow FT$ are in 1-1 correspondence with distributive laws of the monad T over the endofunctor F , i.e., natural transformations $\lambda: TF \Longrightarrow FT$ that respect the monad structure of T . Such distributive laws induce the algebraic structure σ^\sharp uniquely as in (12), with T substituted for Σ .

GSOS specifications. For an example that does not conform to either of the law types explained above, consider the definition of \boxplus in Example 3.2. It does not correspond to a natural transformation $\lambda: \Sigma F \Longrightarrow F\Sigma$ (for $\Sigma X = X \times X$ and $FX = \mathbb{R} \times X$), and intuitively, the reason for this is the use of the variable x_0 on the right side of an equation, or in the target of the conclusion of a rule. However, that definition corresponds to a law of the type

$$\rho: \Sigma(\text{Id} \times F) \Longrightarrow F\Sigma$$

where ρ has X -component:

$$\begin{array}{ccc}
 \rho_X: & (X \times \mathbb{R} \times X) \times (X \times \mathbb{R} \times X) & \Longrightarrow & \mathbb{R} \times (X \times X) \\
 & \langle \langle x_0, a, x_1 \rangle, \langle y_0, b, y_1 \rangle \rangle & \mapsto & \langle a + b, \langle x_0, y_1 \rangle \rangle
 \end{array} \tag{15}$$

One can also combine the expressivity of this type of laws with that of monadic distributive laws described above, and consider the following definition:

Definition 5.1 A *GSOS specification* (of Σ over F) is a natural transformation

$$\rho: \Sigma(\text{Id} \times F) \Longrightarrow FT.$$

◀

GSOS specifications are a particularly useful type of distributive laws, able to describe many interesting definitions. Their name comes from the fact that for the endofunctor $FX = (\mathcal{P}_\omega X)^A$ (see Example 2.6), they correspond (see Turi and Plotkin 1997; Bartels 2004) to structural operational semantic specifications of LTSs in the well known GSOS format (Aceto et al. 2001).

To induce a Σ -operation σ on the final F -coalgebra from a GSOS specification ρ , proceed by analogy with the case of monadic distributive laws and define it as the unique map that makes the diagram:

$$\begin{array}{ccccc}
 \Sigma Z & \xrightarrow{\Sigma(\text{id}, \zeta)} & \Sigma(Z \times FZ) & \xrightarrow{\rho_Z} & FTZ \\
 \sigma \downarrow & & & & \downarrow F\sigma^\sharp \\
 Z & \xrightarrow[\cong]{\zeta} & & & FZ
 \end{array} \tag{16}$$

commute. Again, the unique existence of σ follows from the correspondence of GSOS specifications $\rho: \Sigma(\text{Id} \times F) \Longrightarrow FT$ with distributive laws of the monad T over the copointed endofunctor $(\text{Id} \times F)$, i.e., natural transformations $\lambda: T(\text{Id} \times F) \Longrightarrow (\text{Id} \times F)T$ subject to a few axioms, see (Lenisa et al. 2004) for details. Again, such distributive laws induce the algebraic structure σ^\sharp uniquely as in (12), with T substituted for Σ and $(\text{Id} \times F)$ for F .

6. Pointwise Extensions of Distributive Laws

We shall now focus on the problem of extending distributive laws for arbitrary behaviour functors F to similar laws for $(F-)^A$, so that the resulting operations on final $(F-)^A$ -coalgebras are pointwise extensions of the operations on final F -coalgebras defined by the original distributive laws. It turns out that the solutions applied to the two particular examples in Section 3 work also in this general setting.

6.1. Basic Distributive Laws

As before, let F be a behaviour functor, $\langle Z, \zeta \rangle$ a final F -coalgebra, and Σ an algebraic signature. Suppose that an operation $\sigma: \Sigma Z \rightarrow Z$ arises from a basic distributive law $\lambda: \Sigma F \Longrightarrow F\Sigma$ by coinduction as in (12). Assume moreover that $\langle \bar{Z}, \bar{\zeta} \rangle$ is a final $(F-)^A$ -coalgebra.

Define a basic distributive law $\bar{\lambda}: \Sigma(F-)^A \Longrightarrow (F\Sigma-)^A$ from λ , with $\bar{\lambda}_X$ given as:

$$\Sigma(FX)^A \xrightarrow{\text{cs}_{A,FX}^\Sigma} (\Sigma FX)^A \xrightarrow{\lambda_X^A} (F\Sigma X)^A, \tag{17}$$

and define $\bar{\sigma}: \Sigma \bar{Z} \rightarrow \bar{Z}$ from $\bar{\lambda}$ as in (12).

Theorem 6.1 *Let F be a functor with final coalgebra $\langle Z, \zeta \rangle$, and let $\langle \bar{Z}, \bar{\zeta} \rangle$ be a final F^A -coalgebra. If*

$$\lambda: \Sigma F \Longrightarrow F\Sigma$$

is a distributive law (of Σ over F) which induces a Σ -algebra $\sigma: \Sigma Z \rightarrow Z$, then λ can be lifted to a distributive law

$$\bar{\lambda}: \Sigma(F-)^A \Longrightarrow (F\Sigma-)^A$$

(of Σ over F^A) which induces a Σ -algebra $\bar{\sigma}: \Sigma \bar{Z} \rightarrow \bar{Z}$ that is the pointwise extension of σ .

Proof. Define $\bar{\lambda}$ by (17) and see the Appendix.

Example 6.2 Let us calculate $\bar{\lambda}$ for the law λ in (13) which defines the operation $\oplus: \mathbb{R}^\omega \times \mathbb{R}^\omega \rightarrow \mathbb{R}^\omega$ from Example 3.1. The syntax and behaviour functors are $\Sigma X = X \times X$ and $F^A X = (\mathbb{R} \times X)^\mathbb{R}$, hence an X -component of $\bar{\lambda}$ is of the type:

$$\bar{\lambda}_X: (\mathbb{R} \times X)^\mathbb{R} \times (\mathbb{R} \times X)^\mathbb{R} \Longrightarrow (\mathbb{R} \times X \times X)^\mathbb{R}.$$

Below, for $\phi \in (\mathbb{R} \times X)^\mathbb{R}$ and $a \in \mathbb{R}$, we denote the projections of $\phi(a)$ by $\phi_0(a)$ and $\phi_1(a)$, i.e., $\phi(a) = \langle \phi_0(a), \phi_1(a) \rangle$. Instantiating (17) and (13), $\bar{\lambda}_X$ is defined by:

$$\begin{array}{ccc} \langle \phi, \psi \rangle & \in & (\mathbb{R} \times X)^\mathbb{R} \times (\mathbb{R} \times X)^\mathbb{R} \\ \downarrow & & \downarrow \text{cs}_{\mathbb{R}, \mathbb{R} \times X}^\Sigma \\ \lambda a. \langle \phi_0(a), \phi_1(a), \psi_0(a), \psi_1(a) \rangle & \in & (\mathbb{R} \times X \times \mathbb{R} \times X)^\mathbb{R} \\ \downarrow & & \downarrow \lambda_X^\mathbb{R} \\ \lambda a. \langle \phi_0(a) + \psi_0(a), \langle \phi_1(a), \psi_1(a) \rangle \rangle & \in & (\mathbb{R} \times X \times X)^\mathbb{R} \end{array}$$

Formulating the above $\bar{\lambda}$ as an inference rule we recognise the rule for $\bar{\oplus}$ in (5), with $b = \phi_0(a)$, $c = \psi_0(a)$, $x_1 = \phi_1(a)$ and $y_1 = \psi_1(a)$. \triangleleft

It should be noted that the same construction works for the more expressive type of specifications $\rho: \Sigma F \Longrightarrow FT$. To define the pointwise extension of an operation defined by such a specification, first obtain from ρ a monadic distributive law $\lambda: TF \Longrightarrow FT$, and then apply the construction (17) and Theorem 6.1, with Σ replaced by T throughout.

6.2. GSOS Specifications

We shall now move to define pointwise extensions of operations defined by GSOS specifications $\rho: \Sigma(Id \times F) \Longrightarrow FT$.

First, note that unlike in the case of monadic distributive laws above, one cannot use the correspondence of GSOS specifications and distributive laws of T over $Id \times F$, and apply (17) and Theorem 6.1 with Σ replaced by T and F replaced by $Id \times F$ throughout. The reason for this is that in the natural transformation $\bar{\lambda}$ obtained from (17) in this case, an X -component has domain $T(X \times FX)^A$, and not $T(X \times (FX)^A)$ as required to define the pointwise extension operation by (16).

It seems that the problem can be circumvented by precomposing the counterpart of (17) with an appropriate (co)strength. In this way, from a GSOS specification ρ , one obtains a GSOS specification $\bar{\rho}: \Sigma(Id \times (F-)^A) \Longrightarrow (FT)^A$ with $\bar{\rho}_X$ defined by:

$$\Sigma(X \times (FX)^A) \xrightarrow{\text{cs}_{A, FX}^{\Sigma(X \times -)}} (\Sigma(X \times FX))^A \xrightarrow{\rho_X^A} (FTX)^A, \quad (18)$$

and proceeds to define an extended operation $\bar{\sigma}: \Sigma\bar{Z} \rightarrow \bar{Z}$ from (16).

However, it turns out that $\bar{\sigma}$ obtained this way is *not* the pointwise extension of the operation $\sigma: \Sigma Z \rightarrow Z$, i.e., the relevant diagram (11) does not commute. Indeed, one

may repeat the development of Example 6.2 for (18) with ρ defined by (15), and realise that the resulting $\bar{\rho}$ corresponds to the rule (6), which does not define the pointwise extension of \boxplus as we saw in Example 3.2.

To treat the case of arbitrary GSOS specifications correctly, we proceed in a more subtle way, by analogy with the development of Example 3.2. First, extend the syntax Σ by defining

$$\Sigma_{\triangleright} = A \times - \quad \bar{\Sigma} = \Sigma + \Sigma_{\triangleright}.$$

This amounts to adding $|A|$ auxiliary unary operations to the syntax. In examples, we will denote these operations by $a \triangleright -$, for $a \in A$. Their semantics will intuitively be “one-element buffer” operations. Let \bar{T} be the free monad over $\bar{\Sigma}$.

The pointwise extension of σ will be defined with the help of an algebra $\bar{\sigma} : \bar{\Sigma} \bar{Z} \rightarrow \bar{Z}$ such that the diagram:

$$\begin{array}{ccc} A^\omega \times \Sigma \bar{Z} & \xrightarrow{\text{st}_{A^\omega, \bar{Z}}^\Sigma} & \Sigma(A^\omega \times \bar{Z}) \xrightarrow{\Sigma \text{ev}} \Sigma Z \\ \text{id} \times \iota_{\bar{Z}} \downarrow & & \downarrow \sigma \\ A^\omega \times \bar{\Sigma} \bar{Z} & & \\ \text{id} \times \bar{\sigma} \downarrow & & \\ A^\omega \times \bar{Z} & \xrightarrow{\text{ev}} & \bar{Z} \end{array} \quad (19)$$

commutes (compare (11)), where $\iota : \Sigma \rightarrow \bar{\Sigma}$ is the coproduct injection. To define $\bar{\sigma}$ we provide a GSOS specification $\bar{\rho} : \bar{\Sigma}(Id \times (F-)^A) \Longrightarrow (F\bar{T}-)^A$, by cases of $\bar{\Sigma}$: for Σ_{\triangleright} ,

$$\begin{array}{ccc} A \times (X \times (FX)^A) & \xrightarrow{\pi_{13}} & A \times (FX)^A \xrightarrow{\epsilon_{FX}} FX \\ & & \downarrow \eta_{FX} \\ & & (A \times FX)^A \xrightarrow{(\text{st}_{A, X}^F)^A} (F(A \times X))^A \xrightarrow{} (F\bar{T}X)^A \end{array} \quad (20)$$

and for Σ ,

$$\begin{array}{ccc} \Sigma(X \times (FX)^A) & \xrightarrow{\Sigma(\eta_X \times \text{id})} & \Sigma((A \times X)^A \times (FX)^A) \\ & & \downarrow \text{cs}_{A, A \times X, FX}^{\Sigma(- \times -)} \\ (\Sigma(A \times X \times FX))^A & \xrightarrow{} & (\Sigma((A \times X + X) \times F(A \times X + X)))^A \\ & & \downarrow \rho_{A \times X + X}^A \\ (FT(A \times X + X))^A & \xrightarrow{} & (F\bar{T}X)^A \end{array} \quad (21)$$

This $\bar{\rho}$ defines an algebra $\bar{\sigma} : \bar{\Sigma} \bar{Z} \rightarrow \bar{Z}$ as usual, by (16). With some straightforward, albeit tedious, diagram chasing one shows that (19) commutes:

Theorem 6.3 *Let F be a functor with final coalgebra $\langle Z, \zeta \rangle$, and let $\langle \bar{Z}, \bar{\zeta} \rangle$ be a final*

F^A -coalgebra. If

$$\rho : \Sigma(\text{Id} \times F-) \Longrightarrow (FT-)$$

is a GSOS specification (of Σ over F) which induces a Σ -algebra $\sigma : \Sigma Z \rightarrow Z$, then ρ can be lifted to a GSOS specification

$$\bar{\rho} : \bar{\Sigma}(\text{Id} \times (F-)^A) \Longrightarrow (F\bar{T}-)^A$$

(of $\bar{\Sigma}$ over F^A) which induces a $\bar{\Sigma}$ -algebra $\bar{\sigma} : \bar{\Sigma}\bar{Z} \rightarrow \bar{Z}$ such that $\bar{\sigma} \circ \iota_{\bar{Z}} : \bar{\Sigma}\bar{Z} \rightarrow \bar{Z}$ is the pointwise extension of σ .

Proof. Define $\bar{\rho}$ by (20) and (21) and see the Appendix.

We illustrate the construction behind Theorem 6.3 using the operation \boxplus from Example 3.2.

Example 6.4 Let us calculate $\bar{\rho} : \bar{\Sigma}(\text{Id} \times (F-)^A) \Longrightarrow (F\bar{T}-)^A$ for the law ρ in (15) which defines the operation $\boxplus : \mathbb{R}^\omega \times \mathbb{R}^\omega \rightarrow \mathbb{R}^\omega$. The syntax and behaviour functors are $\Sigma X = X \times X$ and $F^A X = (\mathbb{R} \times X)^\mathbb{R}$, hence $\Sigma_{\triangleright} X = \mathbb{R} \times X$, $\bar{\Sigma} X = (X \times X) + (\mathbb{R} \times X)$ and $\bar{\rho}$ is of type

$$\bar{\rho} : \bar{\Sigma}(\text{Id} \times (\mathbb{R} \times -)^\mathbb{R}) \Longrightarrow (\mathbb{R} \times \bar{T}-)^\mathbb{R}.$$

The two cases for $\bar{\rho}$ have X -components:

$$\begin{aligned} \bar{\rho}_X^\triangleright : \mathbb{R} \times X \times (\mathbb{R} \times X)^\mathbb{R} &\rightarrow (\mathbb{R} \times \bar{T}X)^\mathbb{R} \\ \bar{\rho}_X^\Sigma : \Sigma(X \times (\mathbb{R} \times X)^\mathbb{R}) &\rightarrow (\mathbb{R} \times \bar{T}X)^\mathbb{R} \end{aligned}$$

given by (20) and (21), respectively. Intuitively, $\bar{\rho}^\triangleright$ specifies the F^A -behaviour of the buffer operations and $\bar{\rho}^\Sigma$ specifies the F^A -behaviour of the Σ -operation.

Again, for $\phi \in (\mathbb{R} \times X)^\mathbb{R}$ we let ϕ_0 and ϕ_1 be defined by $\phi(a) = \langle \phi_0(a), \phi_1(a) \rangle$ for all $a \in \mathbb{R}$. To ease the notational burden, we shall suppress some coproduct injections and treat them as set inclusions.

For $\bar{\rho}^\triangleright$, (20) instantiates to:

$$\begin{array}{ccc}
\langle a, x, \phi \rangle & \in & \mathbb{R} \times (X \times (\mathbb{R} \times X)^\mathbb{R}) \\
\downarrow & & \downarrow \pi_{13} \\
\langle a, \phi \rangle & \in & \mathbb{R} \times (\mathbb{R} \times X)^\mathbb{R} \\
\downarrow & & \downarrow \epsilon_{\mathbb{R} \times X} \\
\langle \phi_0(a), \phi_1(a) \rangle & \in & \mathbb{R} \times X \\
\downarrow & & \downarrow \eta_{\mathbb{R} \times X} \\
\lambda c. \langle c, \langle \phi_0(a), \phi_1(a) \rangle \rangle & \in & (\mathbb{R} \times (\mathbb{R} \times X)^\mathbb{R}) \\
\downarrow & & \downarrow (\text{st}_{\mathbb{R}, X}^{\mathbb{R} \times -})^\mathbb{R} \\
\lambda c. \langle \phi_0(a), \langle c, \phi_1(a) \rangle \rangle & \in & (\mathbb{R} \times (\mathbb{R} \times X)^\mathbb{R}) \\
\parallel & & \downarrow \\
\lambda c. \langle \phi_0(a), \langle c, \phi_1(a) \rangle \rangle & \in & (\mathbb{R} \times \overline{TX})^\mathbb{R}
\end{array}$$

The last step involves the inclusion of $\mathbb{R} \times X = \Sigma_\triangleright X$ into \overline{TX} . So $\langle c, \phi_1(a) \rangle$ should be read as $c \triangleright \phi_1(a)$. Formulating $\bar{\rho}^\triangleright$ as a GSOS rule we recognise the rule for $a \triangleright x_0$ in (9) (with $b = \phi_0(a)$, $x_0 = x$ and $x_1 = \phi_1(a)$).

For $\bar{\rho}^\Sigma$, (21) instantiates to:

$$\begin{array}{ccc}
\langle \langle x, \phi \rangle, \langle y, \psi \rangle \rangle & \in & (X \times (\mathbb{R} \times X)^\mathbb{R})^2 \\
\downarrow & & \downarrow \Sigma(\eta_X \times \text{id}) \\
\langle \langle \lambda a. \langle a, x \rangle, \phi \rangle, \langle \lambda a. \langle a, y \rangle, \psi \rangle \rangle & \in & ((\mathbb{R} \times X)^\mathbb{R} \times (\mathbb{R} \times X)^\mathbb{R})^2 \\
\downarrow & & \downarrow \text{cs}_{\mathbb{R}, \mathbb{R} \times X, \mathbb{R} \times X}^{\Sigma(- \times -)} \\
\lambda a. \langle \langle \langle a, x \rangle, \langle \phi_0(a), \phi_1(a) \rangle \rangle, & \in & (((\mathbb{R} \times X) \times (\mathbb{R} \times X))^2)^\mathbb{R} \\
\langle \langle a, y \rangle, \langle \psi_0(a), \psi_1(a) \rangle \rangle \rangle & & \downarrow \\
\parallel & & \downarrow \\
\lambda a. \langle \langle \langle a, x \rangle, \phi_0(a), \phi_1(a) \rangle, & \in & (((\mathbb{R} \times X + X) \times \mathbb{R} \times (\mathbb{R} \times X + X))^2)^\mathbb{R} \\
\langle \langle a, y \rangle, \psi_0(a), \psi_1(a) \rangle \rangle & & \downarrow \rho_{\mathbb{R} \times X + X}^\mathbb{R} \\
\downarrow & & \downarrow \\
\lambda a. \langle \phi_0(a) + \psi_0(a), \langle \langle a, x \rangle, \psi_1(a) \rangle \rangle & \in & (\mathbb{R} \times T(\mathbb{R} \times X + X))^\mathbb{R} \\
\parallel & & \downarrow \\
\lambda a. \langle \phi_0(a) + \psi_0(a), \langle \langle a, x \rangle, \psi_1(a) \rangle \rangle & \in & (\mathbb{R} \times \overline{T(X)})^\mathbb{R}
\end{array}$$

The last step involves the natural inclusion of terms induced by mapping a pair $\langle a, x \rangle \in \mathbb{R} \times X$ to the buffer expression $a \triangleright x \in \Sigma_{\triangleright} X \subseteq \overline{TX}$. With this inclusion in mind, we see that the rule for $\overline{\boxplus}$ in (9) is the rule-based formulation of $\overline{\rho}^{\Sigma}$. \triangleleft

Informally, $\overline{\rho}^{\Sigma}$ can be described as obtained from ρ by replacing all occurrences of $x_0 \in X$ on the right-hand sides with $a \triangleright x_0$ and adding input labels to arrows. In the next section we will see more examples of how $\overline{\rho}$ is obtained from ρ .

7. Further Examples

We shall now sketch a few more example applications of the constructions and results of Section 6.

Example 7.1 Taking $F = 2 \times -$ and $A = 2$, $(2 \times -)$ -behaviours are bitstreams and $(2 \times -)^2$ -coalgebras are binary Mealy machines whose behaviours are causal bitstream functions (cf. Examples 2.1 and 2.2). In (Hansen and Rutten 2010), a coalgebraic treatment was given of the so-called 2-adic bitstream operations and their pointwise extensions to causal bitstream functions. The main purpose in (Hansen and Rutten 2010) was to construct binary Mealy machine realisations from 2-adic function expressions. A crucial step in this result consists of showing that 2-adic function expressions can be given a Mealy machine structure. It was, however, not clear whether this Mealy machine of terms comes about from the existence of a distributive law. A reason for this is that the buffer operations were not explicitly used in (Hansen and Rutten 2010), and so the general picture did not emerge. Since the 2-adic bitstream operations are defined in the GSOS format, Theorem 6.3 tells us that a distributive law exists and how to derive it. We now present this distributive law, except that we leave out the (guarded) inverse operation in order to simplify the presentation. It is straightforward to apply Theorem 6.3 to also include the inverse.

The 2-adic bitstream operations arise from viewing $(a_0, a_1, a_2, \dots) \in 2^{\omega}$ as the coefficients of the power series representation of the 2-adic integer $\sum_{i=0}^{\omega} a_i 2^i$ (cf. Koblitz 1984). The 2-adic integers include the set Q_{odd} of rational numbers with odd denominator via the map $B: Q_{\text{odd}} \rightarrow 2^{\omega}$ which converts numbers to their base 2 representation. For a positive integer n , $B(n)$ is simply obtained by writing n in binary and padding with a tail of zeros, e.g. $B(6) = (0, 1, 1, 0, 0, 0, \dots)$. Negative integers are represented by taking an infinitary version of two's complement, e.g. $B(-6) = (0, 1, 0, 1, 1, 1, \dots)$. More generally, for any $q \in Q_{\text{odd}}$, $B(q)$ is an eventually periodic bitstream.

The 2-adic signature $\Sigma_{2\text{-adic}}$ (without inverse) consists of constants $[0]$, $[1]$ and \mathbf{X} denoting $B(0)$, $B(1)$ and $B(2)$, respectively, together with $+$ (addition), $-$ (unary minus), \times (multiplication). An attractive property of 2-adic representations is that rational arithmetic can be carried out in a sequential manner very similar to how one computes with natural numbers in decimal notation (cf. Hehner and Horspool 1979). The rules below define the 2-adic operations by means of a GSOS specification $\rho: \Sigma(\text{Id} \times 2 \times -) \Longrightarrow 2 \times (T-)$ where T is the free monad generated by $\Sigma_{2\text{-adic}}$. In these rules, \wedge and \oplus denote the bit operations of Boolean AND and addition modulo 2, i.e., for all $a, b \in 2$: $a \wedge b = 1$ iff

$a = b = 1$ and $a \oplus b = 1$ iff $a \neq b$.

$$\begin{array}{c} \overline{[0] \xrightarrow{0} [0]} \quad \overline{[1] \xrightarrow{1} [0]} \quad \overline{\mathbf{X} \xrightarrow{0} [1]} \\ \frac{x_0 \xrightarrow{b} x_1 \quad y_0 \xrightarrow{c} y_1}{x_0 + y_0 \xrightarrow{b \oplus c} (x_1 + y_1) + [b \wedge c]} \quad \frac{x_0 \xrightarrow{b} x_1}{-x_0 \xrightarrow{b} -(x_1 + [b])} \\ \frac{x_0 \xrightarrow{b} x_1 \quad y_0 \xrightarrow{c} y_1}{x_0 \times y_0 \xrightarrow{b \wedge c} (x_1 \times y_0) + ([b] \times y_1)} \end{array}$$

The rule for $+$ shows that carry bits may propagate infinitely to the right, for example $(1, 0, 0, 0, \dots) + (1, 1, 1, \dots) = (0, 0, 0, \dots)$, and the rule for \times shows that multiplication is done in the usual shift-add manner. We refer to (Hansen and Rutten 2010) for more details.

In order to be able to specify bitstream functions rather than just bitstreams, the syntax is extended with a variable \mathbf{s} . The *2-adic function expressions* is thus the set $T(\{\mathbf{s}\})$ of $\Sigma_{2\text{-adic}}$ -terms freely generated over the single variable \mathbf{s} . Algebraically, a term t in $T(\{\mathbf{s}\})$ is evaluated in Γ by interpreting \mathbf{s} as the identity map $\text{id}: 2^\omega \rightarrow 2^\omega$, and the 2-adic operations as their pointwise extensions to Γ . The fact that we can define the pointwise extensions with a distributive law implies that the coalgebraic semantics of 2-adic function expressions coincides with the algebraic semantics. The pointwise extensions of $[0]$, $[1]$, \mathbf{X} , $+$ and $-$ to the binary Mealy functor $(2 \times -)^2$ are obtained from the simple format in (17) and its monadic version. The pointwise extension of multiplication, however, requires the use of the buffer operations. We omit the overline notation and simply write $+$ instead of $\bar{+}$ etc. The typing should be clear from the context.

$$\begin{array}{c} \overline{[0] \xrightarrow{a|0} [0]} \quad \overline{[1] \xrightarrow{a|1} [0]} \quad \overline{\mathbf{X} \xrightarrow{a|0} [1]} \\ \frac{x_0 \xrightarrow{a|b} x_1 \quad y_0 \xrightarrow{a|c} y_1}{x_0 + y_0 \xrightarrow{a|b \oplus c} (x_1 + y_1) + [b \wedge c]} \quad \frac{x_0 \xrightarrow{a|b} x_1}{-x_0 \xrightarrow{a|b} -(x_1 + [b])} \\ \frac{x_0 \xrightarrow{a|b} x_1 \quad y_0 \xrightarrow{a|c} y_1}{x_0 \times y_0 \xrightarrow{a|b \wedge c} (x_1 \times (a \triangleright y_0)) + ([b] \times y_1)} \quad \frac{x_0 \xrightarrow{a|b} x_1}{a \triangleright x_0 \xrightarrow{c|b} c \triangleright x_1} \end{array}$$

Interestingly, it was possible to define the Mealy machine of terms in (Hansen and Rutten 2010), without adding the buffer operations, since these can already be expressed in the existing syntax: for every term $\mathbf{t} \in T(\{\mathbf{s}\})$ and every $a \in 2$, there is a term, denoted by $\mathbf{t}(a:\mathbf{s}) \in T(\{\mathbf{s}\})$, such that for all input streams $\alpha \in 2^\omega$, $\text{beh}(\mathbf{t}(a:\mathbf{s}))(\alpha) = \text{beh}(\mathbf{t})(a:\alpha) = \text{beh}(a \triangleright \mathbf{t})(\alpha)$. We refer to (Hansen and Rutten 2010) for more details on the definition of $\mathbf{t}(a:\mathbf{s})$. \triangleleft

Example 7.2 Consider the functor $FX = 1 + X$ from Example 2.3, that is, F -behaviours are the extended natural numbers $\mathbb{N} + \{\omega\}$, and F^A -behaviours are non-empty, prefix-closed languages (cf. Examples 2.3 and 2.4). The operations \max (\vee), plus ($+$) and times (\cdot) on the extended natural numbers (cf. Example 4.4) are induced by the following

rules:

$$\begin{array}{c}
\frac{x_0 \longrightarrow x_1 \quad y_0 \longrightarrow y_1}{x_0 \vee y_0 \longrightarrow x_1 \vee y_1} \quad \frac{x_0 \longrightarrow x_1 \quad y_0 \longrightarrow \perp}{x_0 \vee y_0 \longrightarrow x_1} \quad \frac{x_0 \longrightarrow \perp \quad y_0 \longrightarrow y_1}{x_0 \vee y_0 \longrightarrow y_1} \\
\\
\frac{x_0 \longrightarrow x_1}{x_0 + y \longrightarrow x_1 + y} \quad \frac{x_0 \longrightarrow \perp \quad y_0 \longrightarrow y_1}{x_0 + y_0 \longrightarrow y_1} \quad \frac{x_0 \longrightarrow \perp \quad y_0 \longrightarrow \perp}{x_0 + y_0 \longrightarrow \perp} \\
\\
\frac{x_0 \longrightarrow x_1 \quad y_0 \longrightarrow y_1}{x_0 \cdot y_0 \longrightarrow (x_1 \cdot y_0) + y_1} \quad \frac{x_0 \longrightarrow \perp}{x_0 \cdot y \longrightarrow \perp} \quad \frac{y_0 \longrightarrow \perp}{x \cdot y_0 \longrightarrow \perp}
\end{array}$$

The pointwise extensions of these operations to the final F^A -coalgebra are defined by the following set of rules. Again, we omit the overline notation for the pointwise extended operations.

$$\begin{array}{c}
\frac{x_0 \xrightarrow{a} x_1}{a \triangleright x_0 \xrightarrow{b} b \triangleright x_1} \quad \frac{x_0 \xrightarrow{a} \perp}{a \triangleright x_0 \xrightarrow{b} \perp} \\
\\
\frac{x_0 \xrightarrow{a} x_1 \quad y_0 \xrightarrow{a} y_1}{x_0 \vee y_0 \xrightarrow{a} x_1 \vee y_1} \quad \frac{x_0 \xrightarrow{a} x_1 \quad y_0 \xrightarrow{a} \perp}{x_0 \vee y_0 \xrightarrow{a} x_1} \quad \frac{x_0 \xrightarrow{a} \perp \quad y_0 \xrightarrow{a} y_1}{x_0 \vee y_0 \xrightarrow{a} y_1} \\
\\
\frac{x_0 \xrightarrow{a} x_1}{x_0 + y \xrightarrow{a} x_1 + (a \triangleright y)} \quad \frac{x_0 \xrightarrow{a} \perp \quad y_0 \xrightarrow{a} y_1}{x_0 + y_0 \xrightarrow{a} y_1} \quad \frac{x_0 \xrightarrow{a} \perp \quad y_0 \xrightarrow{a} \perp}{x_0 + y_0 \xrightarrow{a} \perp} \\
\\
\frac{x_0 \xrightarrow{a} x_1 \quad y_0 \xrightarrow{a} y_1}{x_0 \cdot y_0 \xrightarrow{a} (x_1 \cdot (a \triangleright y_0)) + y_1} \quad \frac{x_0 \xrightarrow{a} \perp}{x_0 \cdot y \xrightarrow{a} \perp} \quad \frac{y_0 \xrightarrow{a} \perp}{x \cdot y_0 \xrightarrow{a} \perp}
\end{array}$$

To illustrate, let $A = \{a, b\}$ and for $n \in \mathbb{N}$, let $\underline{a}^n = \{\varepsilon, a, aa, \dots, a^n\}$, that is, \underline{a}^n denotes the prefix-closed language generated by a^n . We then have for all $n \in \mathbb{N}$,

$$\underline{a}^{n+1} \xrightarrow{a} \underline{a}^n, \quad \underline{a}^0 \xrightarrow{a} \perp, \quad \underline{a}^n \xrightarrow{c} \perp \text{ if } c \neq a.$$

Applying the rule for buffer expressions, we find that for all $c, d \in A$,

$$a \triangleright \underline{a}^{n+1} \xrightarrow{c} c \triangleright \underline{a}^n, \quad c \triangleright a \triangleright \underline{a}^{n+1} \xrightarrow{d} d \triangleright c \triangleright \underline{a}^n, \quad \text{etc.}$$

In general, for all $k \in \mathbb{N}$ and $c_1, \dots, c_k \in A$,

$$\begin{array}{c}
c_{k-1} \dots \triangleright c_1 \triangleright a \triangleright \underline{a}^{n+1} \xrightarrow{c_k} c_k \triangleright c_{k-1} \dots \triangleright c_1 \triangleright \underline{a}^n, \text{ and} \\
c_{k-1} \dots \triangleright c_1 \triangleright a \triangleright \underline{a}^0 \xrightarrow{c_k} \perp
\end{array}$$

Let us check the pointwise extension of $+$ on \underline{a}^3 and \underline{a}^2 . For any stream $\alpha \in A^\omega$ that starts with three a 's, i.e. $\alpha|_3 = a^3$, we have $\text{ev}(\langle \alpha, \underline{a}^3 \rangle) + \text{ev}(\langle \alpha, \underline{a}^2 \rangle) = 3 + 2 = 5$. We now use the above rules to compute the transitions that can be made by $\underline{a}^3 + \underline{a}^2$ on input

$(a, a, a, c_3, c_4, c_5, \dots)$ where $c_i \in A$, $i = 3, 4, 5, \dots$, are arbitrary:

$$\begin{aligned}
\underline{a}^3 + \underline{a}^2 &\xrightarrow{a} \underline{a}^2 + (a \triangleright \underline{a}^2) \\
&\xrightarrow{a} \underline{a}^1 + (a \triangleright a \triangleright \underline{a}^2) \\
&\xrightarrow{a} \underline{a}^0 + (a \triangleright a \triangleright a \triangleright \underline{a}^2) \\
&\xrightarrow{c_3} (c_3 \triangleright a \triangleright a \triangleright \underline{a}^1) \quad (\text{since } \underline{a}^0 \xrightarrow{c_3} \perp) \\
&\xrightarrow{c_4} (c_4 \triangleright c_3 \triangleright a \triangleright \underline{a}^0) \\
&\xrightarrow{c_5} \perp
\end{aligned}$$

We see that $\underline{a}^3 + \underline{a}^2$ can make 5 transitions, so indeed $\text{ev}(\langle \alpha, \underline{a}^3 + \underline{a}^2 \rangle) = \text{ev}(\langle \alpha, \underline{a}^3 \rangle) + \text{ev}(\langle \alpha, \underline{a}^2 \rangle)$ for all α such that $\alpha \upharpoonright_3 = a^3$. If now $\alpha = (a, a, b, c_3, c_4, \dots)$ for arbitrary $c_3, c_4, \dots \in A$, then we find:

$$\begin{aligned}
\underline{a}^3 + \underline{a}^2 &\xrightarrow{a} \underline{a}^2 + (a \triangleright \underline{a}^2) \\
&\xrightarrow{a} \underline{a}^1 + (a \triangleright a \triangleright \underline{a}^2) \\
&\xrightarrow{b} (b \triangleright a \triangleright \underline{a}^1) \quad (\text{since } \underline{a}^1 \xrightarrow{b} \perp) \\
&\xrightarrow{c_3} (c_3 \triangleright b \triangleright \underline{a}^0) \\
&\xrightarrow{c_4} \perp
\end{aligned}$$

Hence $\text{ev}(\langle \alpha, \underline{a}^3 + \underline{a}^2 \rangle) = 4 = 2 + 2 = \text{ev}(\langle \alpha, \underline{a}^3 \rangle) + \text{ev}(\langle \alpha, \underline{a}^2 \rangle)$ as desired. Consider now the product $\underline{a}^3 \cdot \underline{a}^2$ and the input stream a^ω :

$$\begin{aligned}
\underline{a}^3 \cdot \underline{a}^2 &\xrightarrow{a} \underline{a}^2 \cdot (a \triangleright \underline{a}^2) + \underline{a}^1 \\
&\xrightarrow{a} (\underline{a}^1 \cdot (a \triangleright a \triangleright \underline{a}^2) + (a \triangleright \underline{a}^1)) + (a \triangleright \underline{a}^1) \\
&\xrightarrow{a} ((\underline{a}^0 \cdot (a \triangleright a \triangleright a \triangleright \underline{a}^2) + (a \triangleright a \triangleright \underline{a}^1)) + (a \triangleright a \triangleright \underline{a}^1)) + (a \triangleright a \triangleright \underline{a}^1) \\
&\xrightarrow{a} ((a \triangleright a \triangleright \underline{a}^0) + (a \triangleright a \triangleright a \triangleright \underline{a}^1)) + (a \triangleright a \triangleright a \triangleright \underline{a}^1) \\
&\xrightarrow{a} (a \triangleright a \triangleright a \triangleright \underline{a}^0) + (a \triangleright a \triangleright a \triangleright a \triangleright \underline{a}^1) \\
&\xrightarrow{a} (a \triangleright a \triangleright a \triangleright a \triangleright \underline{a}^0) \\
&\xrightarrow{a} \perp
\end{aligned}$$

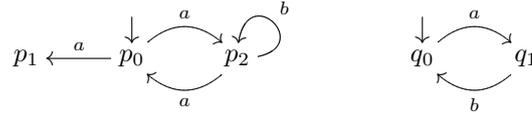
Hence $\text{ev}(\langle a^\omega, \underline{a}^3 \cdot \underline{a}^2 \rangle) = 6 = \text{ev}(\langle a^\omega, \underline{a}^3 \rangle) \cdot \text{ev}(\langle a^\omega, \underline{a}^2 \rangle)$. ◁

Example 7.3 We now look at the case where $F = \mathcal{P}_\omega$. Recall that F -behaviours are strongly extensional, finitely branching trees, and F^A -behaviours are image-finite processes (cf. Examples 2.5 and 2.6). The pointwise extensions to the final F^A -coalgebra of the operations join, composition and interleaving from Example 4.5 are defined by the

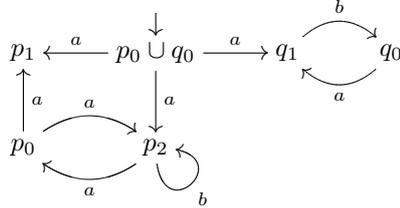
rules below. Again, we omit overlines.

$$\begin{array}{l}
 \text{buffer:} \quad \frac{x_0 \xrightarrow{a} x_1}{a \triangleright x_0 \xrightarrow{b} b \triangleright x_1} \qquad \frac{x_0 \not\xrightarrow{a}}{a \triangleright x_0 \not\xrightarrow{b}} \\
 \\
 \text{join :} \quad \frac{x_0 \xrightarrow{a} x_1}{x_0 \cup y \xrightarrow{a} x_1} \qquad \frac{y_0 \xrightarrow{a} y_1}{x \cup y_0 \xrightarrow{a} y_1} \\
 \\
 \text{composition :} \quad \frac{x_0 \xrightarrow{a} x_1}{x_0; y \xrightarrow{a} x_1; (a \triangleright y)} \qquad \frac{x \not\xrightarrow{a} \quad y_0 \xrightarrow{a} y_1}{x; y_0 \xrightarrow{a} y_1} \\
 \\
 \text{interleaving :} \quad \frac{x_0 \xrightarrow{a} x_1 \quad y_0 \xrightarrow{a} y_1}{x_0 \otimes y_0 \xrightarrow{a} (x_1 \otimes (a \triangleright y_0)) \cup ((a \triangleright x_0) \otimes y_1)}
 \end{array}$$

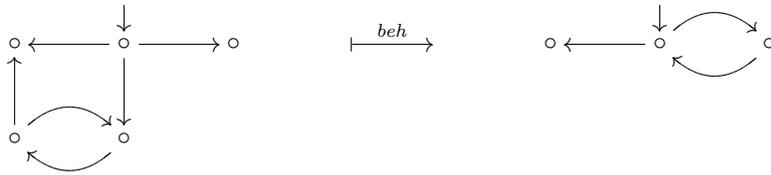
Let us calculate the join of the two processes p_0 and q_0 from Example 4.5. For convenience, their transition diagrams are repeated here (recall that $A = \{a, b\}$):



Applying the rule for \cup to $p_0 \cup q_0$ we find that:



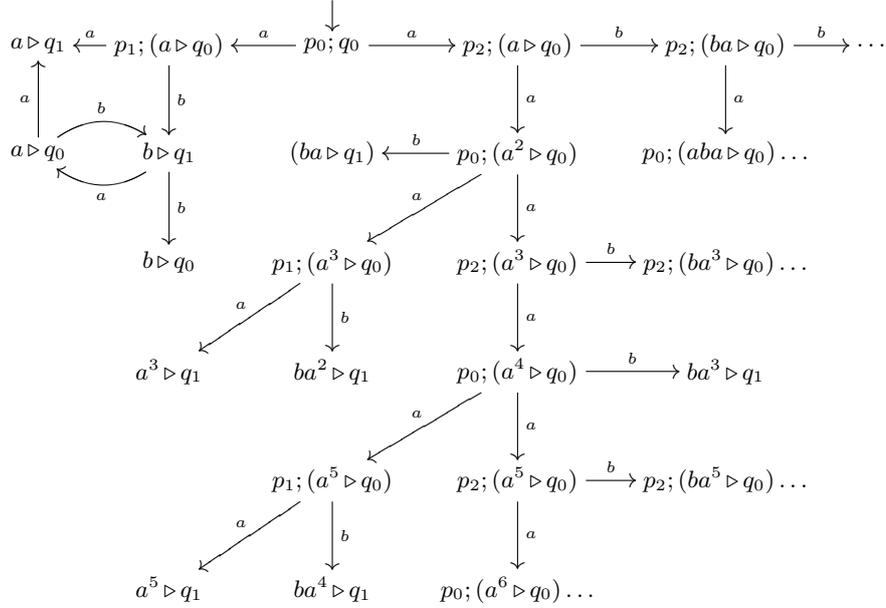
Recall that evaluating a process p on a stream α means restricting to all paths in p that are labelled by a prefix in α , and then dropping the labels. Evaluating $p_0 \cup q_0$ at a^ω , we first find the structure below on the left. Quotienting with bisimilarity we get the structure below on the right



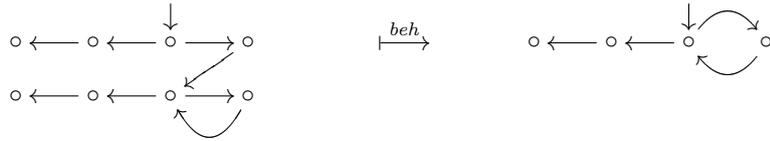
In Example 4.5 we saw that the join of $s_0 = \text{ev}(\langle a^\omega, p_0 \rangle)$ and $x_0 = \text{ev}(\langle a^\omega, q_0 \rangle)$ is equal to s_0 . The above shows that $\text{ev}(\langle a^\omega, p_0 \cup q_0 \rangle) = \text{ev}(\langle a^\omega, p_0 \rangle) \cup \text{ev}(\langle a^\omega, q_0 \rangle) = s_0$.

The join was defined in the simple format, so the buffer operations were not needed. The rules for the composition and the interleaving product, do use the buffers. We now

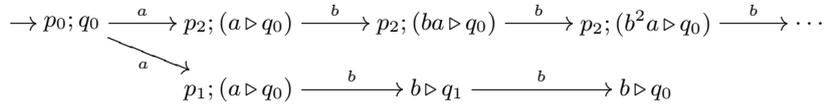
compute part of the transition structure of $p_0; q_0$. In order to keep a compact notation, we will write $w \triangleright x$ instead of $a_1 \triangleright \dots \triangleright a_n \triangleright x$ when $w = a_1 \dots a_n$. Dots after a node indicate that it has outgoing transitions that are not shown.



From the above, it is fairly easy to see that evaluating $p_0; q_0$ at a^ω yields a structure bisimilar to the one below on the left. Quotienting further with bisimilarity, we get the structure on the right, which is equal to $s_0; x_0$ where $s_0 = \text{ev}(\langle a^\omega, p_0 \rangle)$ and $x_0 = \text{ev}(\langle a^\omega, q_0 \rangle)$ (cf. Example 4.5), hence $\text{ev}(\langle a^\omega, p_0; q_0 \rangle) = \text{ev}(\langle a^\omega, p_0 \rangle); \text{ev}(\langle a^\omega, q_0 \rangle)$.



As another example, we compute the transitions of $p_0; q_0$ on ab^ω :



Removing the labels results in a tree which is bisimilar to $t_0; y_0$ where $t_0 = \text{ev}(\langle ab^\omega, p_0 \rangle)$ and $y_0 = \text{ev}(\langle ab^\omega, q_0 \rangle)$ (cf. Example 4.5), and so indeed we have $\text{ev}(\langle ab^\omega, p_0; q_0 \rangle) = \text{ev}(\langle ab^\omega, p_0 \rangle); \text{ev}(\langle ab^\omega, q_0 \rangle)$. Note that the naive approach to defining the operation $;$ on F^A -behaviours would lead to the following behaviour on ab^ω :

$$\begin{array}{c} \rightarrow p_0; q_0 \xrightarrow{a} p_2; q_0 \xrightarrow{b} p_2; q_0 \xrightarrow{b} p_2; q_0 \xrightarrow{b} \dots \\ \searrow^a \\ p_1; q_0 \end{array}$$

If we remove the labels, we obtain a tree which is clearly not bisimilar to $t_0; y_0$. We leave it to the reader to check that for $\alpha \in \{a^\omega, ab^\omega, (ab)^\omega\}$, $\text{ev}(\langle \alpha, p_0 \otimes q_0 \rangle) = \text{ev}(\langle \alpha, p_0 \rangle) \otimes \text{ev}(\langle \alpha, q_0 \rangle)$. \triangleleft

8. Conclusions and Future Work

We have shown how GSOS rules that define operations on F -behaviours can be lifted to GSOS rules that define the corresponding pointwise extensions on F^A -behaviours. The construction is carried out uniformly for all behaviour functors F and it relies on extending the signature with a family of buffer operations. Although the proof is technically involved, applying the construction to concrete examples is straightforward. Our original motivating example was the pointwise extension of the 2-adic operations on bitstreams to causal bitstream functions (Mealy machine behaviours) from (Hansen and Rutten 2010). In this example, a specification language for bitstreams is turned into one for bitstream functions by adding a variable to the syntax as in standard mathematical practice. We expect that this ‘trick’ is useful in other settings. In this perspective, our result can be formulated as: a structural operational semantics for Σ -expressions can be systematically lifted to one for Σ -function expressions in such a way that the operational (coalgebraic) semantics coincides with the algebraic semantics.

As related work we mention (Jacobs 2006), where it is shown that an Eilenberg-Moore algebra $\sigma: TB \rightarrow B$ for a monad T induces a distributive law of T over the (Moore) functor $FX = B \times X^A$, and the Eilenberg-Moore algebra induced on the carrier B^{A^+} of the final F -coalgebra is the pointwise extension of σ . Due to the shape of the functor $FX = B \times X^A$, our present result does not directly relate to that in (Jacobs 2006), however, it is straightforward to state and prove an analogous one: If $\sigma: TB \rightarrow B$ is an Eilenberg-Moore algebra, then σ gives rise to a distributive law of T over the Mealy functor $(B \times -)^A$ such that the induced Eilenberg-Moore algebra $\hat{\sigma}: T(B^{A^+}) \rightarrow B^{A^+}$ is the pointwise extension of σ . Note that causal stream functions $A^\omega \rightarrow B^\omega$ are in 1-1 correspondence with functions $A^+ \rightarrow B$. In our setting, we can describe the pointwise extension of σ in two steps. First, from σ it is easy to define a simple distributive law λ of T over the functor $(B \times -)$. The element-wise addition of streams over \mathbb{R} is such an example. This λ defines an operation σ^ω on B^ω which is the pointwise (i.e. element-wise) extension of σ . Second, we apply Theorem 6.1 to lift λ and obtain the pointwise extension $\overline{\sigma^\omega}$ on Γ . One easily shows that $\overline{\sigma^\omega}$ corresponds to $\hat{\sigma}$ under the bijection $\Gamma \cong B^{A^+}$.

As mentioned in Section 4, the input evaluation $s \times m$ is a construction very similar to the wreath product of automata (cf. Carton 2000; Pin and Weil 2002; Straubing 1989). The wreath product exists for different types of automata, but the basic idea in all of them is that the output of the one automaton becomes the input of the other. Several results in formal language theory have been proved using the wreath product. For

example, taking the wreath product of a finite deterministic automaton and a sequential transducer, one easily concludes that sequential functions preserve regularity of languages under inverse images. Deeper results formulated in terms of decomposition of semigroups include the Krohn-Rhodes theorem and characterisations of concatenation hierarchies. We refer to (cf. Carton 2000; Pin and Weil 2002; Straubing 1989) for further references to the literature. It would be interesting to study wreath products more generally in a coalgebraic setting, and explore their applicability in the coalgebraic theory of formal languages (cf. Rutten 2003).

Another observation of the \ltimes -operation, is that it suggests a characterisation of F^A -behaviours as causal functions f from A^ω to F -behaviours. Here, we call f causal if for all $n \in \mathbb{N}$, $\alpha|_n = \beta|_n$ implies that $f(\alpha)$ and $f(\beta)$ are n -step bisimilar. We leave the details of such a characterisation for future work.

Appendix A. Proofs

A.1. Preliminaries

We begin with some basic results and definitions that will be useful in the following.

For any functor F , strength st^F and costrength cs^F (see Section 2.1) correspond to each other via the adjunction $A \times - \dashv (-)^A$. In particular, there is a commuting diagram:

$$\begin{array}{ccc} A \times F(X^A) & \xrightarrow{\text{st}_{A,X^A}^F} & F(A \times X^A) \\ \text{id} \times \text{cs}_{A,X}^F \downarrow & & \downarrow F\epsilon_X \\ A \times (FX)^A & \xrightarrow{\epsilon_{FX}} & FX. \end{array} \quad (22)$$

Strength respects products: there is

$$\text{st}_{A \times B, X}^F = \text{st}_{A, B \times X}^F \circ (\text{id}_A \times \text{st}_{B, X}^F). \quad (23)$$

Strength transformations compose along functor composition: for endofunctors F and G on **Set**, there is

$$\text{st}_{A, X}^{GF} = G \text{st}_{A, X}^F \circ \text{st}_{A, FX}^G \quad (24)$$

for any A and X . Strength is natural also in the endofunctor F . Indeed, for any natural transformation $\alpha : F \Rightarrow G$, there is:

$$\begin{array}{ccc} A \times FX & \xrightarrow{\text{st}_{A, X}^F} & F(A \times X) \\ \text{id} \times \alpha_X \downarrow & & \downarrow \alpha_{A \times X} \\ A \times GX & \xrightarrow{\text{st}_{A, X}^G} & G(A \times X). \end{array} \quad (25)$$

To simplify the notation, we will sometimes use a natural transformation $e : A^\omega \times (-)^A \Rightarrow A^\omega \times -$, with the component e_X defined by:

$$A^\omega \times X^A \xrightarrow{\langle \text{tl}, \text{hd} \rangle \times \text{id}} A^\omega \times A \times X^A \xrightarrow{\text{id} \times \epsilon_X} A^\omega \times X. \quad (26)$$

With this, the definition (10) of the map ev can be rewritten as:

$$\begin{array}{ccc}
 A^\omega \times \bar{Z} & \xrightarrow{\text{ev}} & Z \\
 \text{id} \times \bar{\zeta} \downarrow & & \downarrow \zeta \\
 A^\omega \times (F\bar{Z})^A & & \\
 e_{F\bar{Z}} \downarrow & & \\
 A^\omega \times F\bar{Z} & & \\
 \text{st}_{A^\omega, \bar{Z}}^F \downarrow & & \\
 F(A^\omega \times \bar{Z}) & \xrightarrow{F\text{ev}} & FZ.
 \end{array} \tag{27}$$

A.2. Proof of Theorem 6.1

Recall that $\bar{\lambda}$ is defined by (17), and $\bar{\sigma}$ is defined from $\bar{\lambda}$ as in (12), specifically by:

$$\begin{array}{ccc}
 \Sigma\bar{Z} & \xrightarrow{\Sigma\bar{\zeta}} & \Sigma(F\bar{Z})^A & \xrightarrow{\bar{\lambda}_{\bar{Z}}} & (F\Sigma\bar{Z})^A \\
 \bar{\sigma} \downarrow \text{dotted} & & & & \downarrow (F\bar{\sigma})^A \\
 \bar{Z} & \xrightarrow[\cong]{\bar{\zeta}} & (F\bar{Z})^A & &
 \end{array} \tag{28}$$

We shall prove that (11) commutes. To this end, define an F -coalgebra structure on $A^\omega \times \Sigma\bar{Z}$:

$$\begin{array}{ccc}
 A^\omega \times \Sigma\bar{Z} & \xrightarrow{\text{id} \times \Sigma\bar{\zeta}} & A^\omega \times \Sigma(F\bar{Z})^A & \xrightarrow{\text{id} \times \bar{\lambda}_{\bar{Z}}} & A^\omega \times (F\Sigma\bar{Z})^A \\
 & & & & \downarrow e_{F\Sigma\bar{Z}} \\
 & & & & A^\omega \times F\Sigma\bar{Z} & \xrightarrow{\text{st}_{A^\omega, \Sigma\bar{Z}}^F} & F(A^\omega \times \Sigma\bar{Z})
 \end{array}$$

and check that both sides of (11) are coalgebra morphisms from it to the final F -

coalgebra. For the southwestern side, check the diagram:

$$\begin{array}{ccccc}
 A^\omega \times \Sigma \bar{Z} & \xrightarrow{\text{id} \times \bar{\sigma}} & A^\omega \times \bar{Z} & \xrightarrow{\text{ev}} & Z \\
 \text{id} \times \Sigma \bar{\zeta} \downarrow & & \downarrow \text{id} \times \bar{\zeta} & & \downarrow \zeta \\
 A^\omega \times \Sigma (F\bar{Z})^A & \xrightarrow{\quad (i) \quad} & A^\omega \times (F\bar{Z})^A & & \\
 \text{id} \times \bar{\lambda}_{\bar{Z}} \downarrow & & \downarrow e_{F\bar{Z}} & & \\
 A^\omega \times (F\Sigma\bar{Z})^A & \xrightarrow{\text{id} \times (F\bar{\sigma})^A} & A^\omega \times (F\bar{Z})^A & & \\
 e_{F\Sigma\bar{Z}} \downarrow & \text{(iii)} & \downarrow e_{F\bar{Z}} & & \\
 A^\omega \times F\Sigma\bar{Z} & \xrightarrow{\text{id} \times F\bar{\sigma}} & A^\omega \times F\bar{Z} & & \\
 \text{st}_{A^\omega, \Sigma\bar{Z}}^F \downarrow & \text{(iv)} & \downarrow \text{st}_{A^\omega, \bar{Z}}^F & & \\
 F(A^\omega \times \Sigma\bar{Z}) & \xrightarrow{F(\text{id} \times \bar{\sigma})} & F(A^\omega \times \bar{Z}) & \xrightarrow{F\text{ev}} & FZ
 \end{array}$$

where (i) commutes by (28), (ii) by (27), (iii) by naturality of e and (iv) by naturality of st^F .

For the northeastern side of (11), in the diagram:

$$\begin{array}{ccccccc}
 A^\omega \times \Sigma \bar{Z} & \xrightarrow{\text{st}_{A^\omega, \bar{Z}}^\Sigma} & \Sigma(A^\omega \times \bar{Z}) & \xrightarrow{\Sigma\text{ev}} & \Sigma Z & \xrightarrow{\sigma} & Z \\
 \text{id} \times \Sigma \bar{\zeta} \downarrow & & \downarrow \Sigma(\text{id} \times \bar{\zeta}) & & \downarrow \Sigma\zeta & & \downarrow \zeta \\
 A^\omega \times \Sigma (F\bar{Z})^A & \xrightarrow{\text{st}_{A^\omega, (F\bar{Z})^A}^\Sigma} & \Sigma(A^\omega \times (F\bar{Z})^A) & & \downarrow \Sigma e_{F\bar{Z}} & & \\
 \text{id} \times \bar{\lambda}_{\bar{Z}} \downarrow & \text{id} \times \text{cs}_{A, F\bar{Z}}^\Sigma \searrow & A^\omega \times (\Sigma F\bar{Z})^A & & \downarrow \Sigma e_{F\bar{Z}} & & \\
 \text{id} \times \bar{\lambda}_{\bar{Z}} \downarrow & \text{id} \times \lambda_{\bar{Z}}^A \searrow & A^\omega \times (\Sigma F\bar{Z})^A & & \downarrow e_{\Sigma F\bar{Z}} & & \\
 A^\omega \times (F\Sigma\bar{Z})^A & & A^\omega \times \Sigma F\bar{Z} & \xrightarrow{\text{st}_{A^\omega, F\bar{Z}}^\Sigma} & \Sigma(A^\omega \times F\bar{Z}) & & \\
 e_{F\Sigma\bar{Z}} \downarrow & \text{id} \times \lambda_{\bar{Z}} \swarrow & A^\omega \times \Sigma F\bar{Z} & & \downarrow \Sigma \text{st}_{A^\omega, \bar{Z}}^F & & \\
 A^\omega \times F\Sigma\bar{Z} & & A^\omega \times \Sigma F\bar{Z} & & \Sigma F(A^\omega \times \bar{Z}) & \xrightarrow{\Sigma F\text{ev}} & \Sigma FZ \\
 \text{st}_{A^\omega, \Sigma\bar{Z}}^F \downarrow & & \downarrow \lambda_{A^\omega \times \bar{Z}} & & \downarrow \lambda_{A^\omega \times \bar{Z}} & & \\
 F(A^\omega \times \Sigma\bar{Z}) & \xrightarrow{F \text{st}_{A^\omega, \bar{Z}}^\Sigma} & F\Sigma(A^\omega \times \bar{Z}) & \xrightarrow{F\Sigma\text{ev}} & F\Sigma Z & \xrightarrow{F\sigma} & FZ
 \end{array}$$

(i) commutes by naturality of st^Σ , (ii) by (27), (iii) by (12), (iv) by naturality of λ , (v) follows from (22) and (26) using naturality of st^Σ and (23), and (vi) follows from (24)–(25); the remaining parts commute due to naturality of e and the definition of $\bar{\lambda}$ in (17). QED

A.3. Proof of Theorem 6.3

We shall now prove that (19) commutes, recalling that ev is defined by (27) and $\bar{\sigma}$ is defined from $\bar{\rho}$ as in (16), specifically by:

$$\begin{array}{ccc} \bar{\Sigma} \bar{Z} & \xrightarrow{\bar{\Sigma}(\text{id}, \bar{\zeta})} & \bar{\Sigma}(\bar{Z} \times (F\bar{Z})^A) \xrightarrow{\bar{\rho}_{\bar{Z}}} (F\bar{T} \bar{Z})^A \\ \downarrow \bar{\sigma} & & \downarrow (F\bar{\sigma}^\#)^A \\ \bar{Z} & \xrightarrow[\cong]{\bar{\zeta}} & (F\bar{Z})^A, \end{array} \quad (29)$$

where $\bar{\rho}$ is defined by (20) and (21).

In the proof, we shall use an auxiliary natural transformation

$$\theta : A^\omega \times \bar{\Sigma} - \Longrightarrow (\Sigma + \text{Id})(A^\omega \times -)$$

with θ_X defined by cases of $\bar{\Sigma}$: for Σ_{\triangleright} ,

$$A^\omega \times A \times X \xrightarrow{\langle \text{tl}, \text{hd} \rangle^{-1} \times \text{id}} A^\omega \times X \hookrightarrow (\Sigma + \text{Id})(A^\omega \times X)$$

and for Σ ,

$$A^\omega \times \Sigma X \xrightarrow{\text{st}_{A^\omega, X}^\Sigma} \Sigma(A^\omega \times X) \hookrightarrow (\Sigma + \text{Id})(A^\omega \times X).$$

The diagram (19) can be decomposed as:

$$\begin{array}{ccccc} A^\omega \times \Sigma \bar{Z} & \xrightarrow{\text{st}_{A^\omega, \bar{Z}}^\Sigma} & \Sigma(A^\omega \times \bar{Z}) & \xrightarrow{\Sigma \text{ev}} & \Sigma Z \\ \text{id} \times \iota_{\bar{Z}} \downarrow & & \downarrow & & \downarrow \\ A^\omega \times \bar{\Sigma} \bar{Z} & \xrightarrow{\theta_{\bar{Z}}} & \Sigma(A^\omega \times \bar{Z}) + (A^\omega \times \bar{Z}) & \xrightarrow{\Sigma \text{ev} + \text{ev}} & \Sigma Z + Z \\ \text{id} \times \bar{\sigma} \downarrow & & & & \downarrow [\sigma, \text{id}] \\ A^\omega \times \bar{Z} & \xrightarrow{\text{ev}} & & & Z, \end{array} \quad (30)$$

where the top left square commutes by the second clause of definition of θ , and the top right square commutes obviously. In the following we shall prove that the bottom rectangle commutes as well.

To this end, consider a map $\gamma : A^\omega \times \bar{\Sigma} \bar{Z} \rightarrow F(A^\omega \times \bar{T} \bar{Z})$ defined by:

$$\begin{array}{ccc} A^\omega \times \bar{\Sigma} \bar{Z} & \xrightarrow{\text{id} \times \bar{\Sigma}(\text{id}, \bar{\zeta})} & A^\omega \times \bar{\Sigma}(\bar{Z} \times (F\bar{Z})^A) \xrightarrow{\text{id} \times \bar{\rho}_{\bar{Z}}} A^\omega \times (F\bar{T} \bar{Z})^A \\ & & \downarrow e_{F\bar{T} \bar{Z}} \\ & & A^\omega \times F\bar{T} \bar{Z} \xrightarrow{\text{st}_{A^\omega, \bar{T} \bar{Z}}^F} F(A^\omega \times \bar{T} \bar{Z}). \end{array}$$

Note that in this definition all maps except the first one are natural in \bar{Z} . This allows us to use a definition/proof principle similar to that used in (14), and infer that both sides of the lower part of (30) are equal if they are both “maps” from γ to the final

F -coalgebra ζ in the following sense:

$$\begin{array}{ccccc}
A^\omega \times \overline{\Sigma} \overline{Z} & \xrightarrow{\text{id} \times \overline{\sigma}} & A^\omega \times \overline{Z} & \xrightarrow{\text{ev}} & Z \\
\downarrow \gamma & & & & \downarrow \zeta \\
F(A^\omega \times \overline{T} \overline{Z}) & \xrightarrow{F(\text{id} \times \overline{\sigma}^\sharp)} & F(A^\omega \times \overline{Z}) & \xrightarrow{F\text{ev}} & FZ
\end{array} \quad (31)$$

$$\begin{array}{ccccc}
A^\omega \times \overline{\Sigma} \overline{Z} & \xrightarrow{\theta_{\overline{Z}}} & \Sigma(A^\omega \times \overline{Z}) + (A^\omega \times \overline{Z}) & \xrightarrow{\Sigma\text{ev} + \text{ev}} & \Sigma Z + Z & \xrightarrow{[\sigma, \text{id}]} & Z \\
\downarrow \gamma & & & & & & \downarrow \zeta \\
F(A^\omega \times \overline{T} \overline{Z}) & \xrightarrow{F\theta_{\overline{Z}}^\sharp} & FT(A^\omega \times \overline{Z}) & \xrightarrow{FT\text{ev}} & FTZ & \xrightarrow{F\sigma^\sharp} & FZ
\end{array} \quad (32)$$

where $\theta^\sharp : A^\omega \times \overline{T} - \implies T(A^\omega \times -)$ is the obvious inductive extension of θ . The above two conditions can equivalently be understood as two coalgebra morphism diagrams to ζ from an F -coalgebra on $A^\omega \times \overline{T} \overline{Z}$ easily defined from γ ; however, they are a bit easier to prove in the present formulation.

The condition (31) is proved by analogy with the corresponding condition in the proof of Theorem 6.1; chase the diagram:

$$\begin{array}{ccccc}
A^\omega \times \overline{\Sigma} \overline{Z} & \xrightarrow{\text{id} \times \overline{\sigma}} & A^\omega \times \overline{Z} & \xrightarrow{\text{ev}} & Z \\
\downarrow \text{id} \times \overline{\Sigma}(\text{id}, \overline{\zeta}) & & \downarrow \text{id} \times \overline{\zeta} & & \downarrow \zeta \\
A^\omega \times \overline{\Sigma} \overline{\Sigma}(\overline{Z} \times (F\overline{Z})^A) & \xrightarrow{\text{(i)}} & A^\omega \times \overline{\Sigma} \overline{\Sigma}(\overline{Z} \times (F\overline{Z})^A) & & \\
\downarrow \text{id} \times \overline{\rho}_{\overline{Z}} & & \downarrow \text{id} \times (F\overline{\sigma}^\sharp)^A & & \\
A^\omega \times (F\overline{T} \overline{Z})^A & \xrightarrow{\text{id} \times (F\overline{\sigma}^\sharp)^A} & A^\omega \times (F\overline{Z})^A & & \\
\downarrow e_{F\overline{T} \overline{Z}} & & \downarrow e_{F\overline{Z}} & & \\
A^\omega \times F\overline{T} \overline{Z} & \xrightarrow{\text{id} \times F\overline{\sigma}^\sharp} & A^\omega \times F\overline{Z} & & \\
\downarrow \text{st}_{A^\omega, \overline{T} \overline{Z}}^F & & \downarrow \text{st}_{A^\omega, \overline{Z}}^F & & \\
F(A^\omega \times \overline{T} \overline{Z}) & \xrightarrow{F(\text{id} \times \overline{\sigma}^\sharp)} & F(A^\omega \times \overline{Z}) & \xrightarrow{F\text{ev}} & FZ
\end{array}$$

where (i) commutes by (29), (ii) by (27), (iii) by naturality of e and (iv) by naturality of st^F .

The condition (32) requires more care, and is best proved by case analysis of $\bar{\Sigma}$. Unfold the definition of $\bar{\rho}_{\bar{Z}}$ in the definition of γ , and recalling the definition of θ by cases, for Σ_{\triangleright} chase the diagram:

$$\begin{array}{ccccc}
 A^\omega \times A \times \bar{Z} & \xrightarrow{\langle tl, hd \rangle^{-1} \times id} & A^\omega \times \bar{Z} & \xrightarrow{ev} & Z \\
 id \times \bar{\zeta} \downarrow & (i) & \downarrow id \times \bar{\zeta} & & \downarrow \zeta \\
 A^\omega \times A \times (F\bar{Z})^A & \xrightarrow{\langle tl, hd \rangle^{-1} \times id} & A^\omega \times (F\bar{Z})^A & & \\
 id \times \epsilon_{F\bar{Z}} \downarrow & & \downarrow e_{F\bar{Z}} & & \\
 A^\omega \times F\bar{Z} & (ii) & & & \\
 id \times \eta_{F\bar{Z}} \downarrow & & \downarrow id & & \\
 A^\omega \times (A \times F\bar{Z})^A & & & & \\
 id \times (st_{A, \bar{Z}}^F)^A \swarrow & & \downarrow e_{A \times F\bar{Z}} & (iii) & \downarrow e_{F\bar{Z}} \\
 A^\omega \times (F(A \times \bar{Z}))^A & (v) & A^\omega \times A \times F\bar{Z} & \xrightarrow{\langle tl, hd \rangle^{-1} \times id} & A^\omega \times F\bar{Z} \\
 e_{F(A \times \bar{Z})} \downarrow & & \downarrow id \times st_{A, \bar{Z}}^F & (vi) & \downarrow st_{A^\omega, \bar{Z}}^F \\
 A^\omega \times F(A \times \bar{Z}) & & & & \\
 st_{A^\omega, A \times \bar{Z}}^F \downarrow & & & & \\
 F(A^\omega \times A \times \bar{Z}) & \xrightarrow{F(\langle tl, hd \rangle^{-1} \times id)} & F(A^\omega \times \bar{Z}) & \xrightarrow{Fev} & FZ
 \end{array}$$

where (i) commutes trivially, (ii) and (iii) by definition of e , (iv) by (27), (v) by naturality of e and (vi) by (23) and naturality of st^F .

On the other hand, for Σ , chase the diagram:

$$\begin{array}{ccccc}
A^\omega \times \Sigma \bar{Z} & \xrightarrow{\text{st}_{A^\omega, \bar{Z}}^\Sigma} & \Sigma(A^\omega \times \bar{Z}) & \xrightarrow{\Sigma \text{ev}} & \Sigma Z \xrightarrow{\sigma} Z \\
\text{id} \times \Sigma(\text{id}, \bar{\zeta}) \downarrow & & \downarrow \Sigma(\text{id} \times (\text{id}, \bar{\zeta})) & & \downarrow \Sigma(\text{id}, \zeta) \\
A^\omega \times \Sigma(\bar{Z} \times (F\bar{Z})^A) & \xrightarrow{\text{st}_{\dots}^\Sigma} & \Sigma(A^\omega \times \bar{Z} \times (F\bar{Z})^A) & & \downarrow \zeta \\
\text{id} \times \Sigma(\eta_{\bar{Z}} \times \text{id}) \downarrow & & \downarrow \Sigma(\text{id} \times \eta_{\bar{Z}} \times \text{id}) & & \\
A^\omega \times \Sigma((A \times \bar{Z})^A \times (F\bar{Z})^A) & \xrightarrow{\text{st}_{\dots}^\Sigma} & \Sigma(A^\omega \times (A \times \bar{Z})^A \times (F\bar{Z})^A) & & \\
\text{id} \times \text{cs}_{A, A \times \bar{Z}, F\bar{Z}}^{\Sigma(- \times -)} \downarrow & & \downarrow \Sigma(\text{id} \times \text{cs}_{A \times \bar{Z}, F\bar{Z}}^{\Sigma(- \times -)}) & & \\
A^\omega \times (\Sigma(A \times \bar{Z} \times F\bar{Z}))^A & & \Sigma(A^\omega \times (A \times \bar{Z} \times F\bar{Z})^A) & & \\
e_{\Sigma(A \times \bar{Z} \times F\bar{Z})} \downarrow & & \downarrow \Sigma e_{A \times \bar{Z} \times F\bar{Z}} & & \\
A^\omega \times \Sigma(A \times \bar{Z} \times F\bar{Z}) & \xrightarrow{\text{st}_{\dots}^\Sigma} & \Sigma(A^\omega \times A \times \bar{Z} \times F\bar{Z}) & & \\
\text{id} \times \rho_{A \times \bar{Z} + \bar{Z}} \downarrow & & \downarrow \Sigma(\Delta \times \text{id}) & & \\
A^\omega \times FT(A \times \bar{Z} + \bar{Z}) & & \Sigma(A^\omega \times A \times \bar{Z} \times A^\omega \times F\bar{Z}) & & \\
\downarrow & & \downarrow \Sigma((tl, hd)^{-1} \times \text{id} \times \text{st}_{A^\omega, \bar{Z}}^F) & & \\
A^\omega \times FT\bar{Z} & & \Sigma(A^\omega \times \bar{Z} \times F(A^\omega \times \bar{Z})) & \xrightarrow{\Sigma(\text{ev} \times F\text{ev})} & \Sigma(Z \times FZ) \\
\text{st}_{A^\omega, \bar{T}\bar{Z}}^F \downarrow & & \downarrow \rho_{A^\omega \times \bar{Z}} & & \downarrow \rho_Z \\
F(A^\omega \times \bar{T}\bar{Z}) & \xrightarrow{F\theta_{\bar{Z}}^\sharp} & FT(A^\omega \times \bar{Z}) & \xrightarrow{FT\text{ev}} & FTZ \xrightarrow{F\sigma^\sharp} FZ
\end{array} \quad (33)$$

where in the top right corner of (iv), $\Delta : A^\omega \rightarrow A^\omega \times A^\omega$ is the diagonal map, with rearrangement of products left implicit. Here (i) and (ii) commute by naturality of st^Σ , (vi) by naturality of ρ and (vii) by (14). It remains to be shown that (iii)–(v) commute as well.

(iii) in (33) commutes. First, for any X , the diagram

$$\begin{array}{ccc}
A^\omega \times \Sigma X^A & \xrightarrow{\text{st}_{A^\omega, X^A}^\Sigma} & \Sigma(A^\omega \times X^A) \\
\text{id} \times \text{cs}_{A, X}^\Sigma \downarrow & & \downarrow \Sigma e_X \\
A^\omega \times (\Sigma X)^A & & \downarrow \Sigma e_X \\
e_{\Sigma X} \downarrow & & \downarrow \\
A^\omega \times \Sigma X & \xrightarrow{\text{st}_{A^\omega, X}^\Sigma} & \Sigma(A^\omega \times X)
\end{array}$$

commutes by (22), by the definition (26) of e and by (23). For $X = U \times V$, this becomes

the bottom part of

$$\begin{array}{ccc}
A^\omega \times \Sigma(U^A \times V^A) & \xrightarrow{\text{st}_{A^\omega, U^A \times V^A}^\Sigma} & \Sigma(A^\omega \times U^A \times V^A) \\
\text{id} \times \Sigma \text{cs}_{U, V}^\times \downarrow & & \downarrow \Sigma(\text{id} \times \text{cs}_{U, V}^\times) \\
A^\omega \times \Sigma(U \times V)^A & \xrightarrow{\text{st}_{A^\omega, (U \times V)^A}^\Sigma} & \Sigma(A^\omega \times (U \times V)^A) \\
\text{id} \times \text{cs}_{A, U \times V}^\Sigma \downarrow & & \downarrow \Sigma e_{U \times V} \\
A^\omega \times (\Sigma(U \times V))^A & & \\
e_{\Sigma(U \times V)} \downarrow & & \\
A^\omega \times \Sigma(U \times V) & \xrightarrow{\text{st}_{A^\omega, U \times V}^\Sigma} & \Sigma(A^\omega \times (U \times V)),
\end{array}$$

where the top part commutes by naturality of st^Σ . Substituting $U = A \times \bar{Z}$ and $V = F\bar{Z}$, we obtain (iii) of (33).

(iv) in (33) commutes. Denoting $\hat{F} = Id \times F$ and defining a function $\phi : A^\omega \times (A \times \bar{Z} + \bar{Z}) \rightarrow A^\omega \times \bar{Z}$ by:

$$A^\omega \times (A \times \bar{Z} + \bar{Z}) \xrightarrow{\cong} A^\omega \times A \times \bar{Z} + A^\omega \times \bar{Z} \xrightarrow{[(tl, hd)^{-1}, \text{id}]} A^\omega \times \bar{Z},$$

chase the diagram

$$\begin{array}{ccccc}
A^\omega \times \Sigma(A \times \bar{Z} \times F\bar{Z}) & \xrightarrow{\text{st}_{A^\omega, A \times \bar{Z} \times F\bar{Z}}^\Sigma} & \Sigma(A^\omega \times A \times \bar{Z} \times F\bar{Z}) & & \\
\downarrow & \searrow & \downarrow \Sigma(\Delta \times \text{id}) & & \\
& & \Sigma(A^\omega \times A \times \bar{Z} \times A^\omega \times F\bar{Z}) & & \\
& & \downarrow \Sigma(\text{id} \times \text{st}_{A^\omega, \bar{Z}}^F) & & \\
& & \Sigma(A^\omega \times A \times \bar{Z} \times F(A^\omega \times \bar{Z})) & & \\
& & \downarrow \Sigma((tl, hd)^{-1} \times \text{id}) & & \\
& & \Sigma(A^\omega \times \bar{Z}) & & \\
& & \downarrow \rho_{A^\omega \times \bar{Z}} & & \\
A^\omega \times \Sigma \hat{F}(A \times \bar{Z} + \bar{Z}) & \xrightarrow{\text{st}_{A^\omega, A \times \bar{Z} + \bar{Z}}^{\Sigma \hat{F}}} & \Sigma \hat{F}(A^\omega \times (A \times \bar{Z} + \bar{Z})) & \xrightarrow{\Sigma \hat{F} \phi} & \Sigma \hat{F}(A^\omega \times \bar{Z}) \\
\downarrow \text{id} \times \rho_{A \times \bar{Z} + \bar{Z}} & & \downarrow \rho_{A^\omega \times (A \times \bar{Z} + \bar{Z})} & & \downarrow \rho_{A^\omega \times \bar{Z}} \\
A^\omega \times FT(A \times \bar{Z} + \bar{Z}) & \xrightarrow{\text{st}_{A^\omega, A \times \bar{Z} + \bar{Z}}^{FT}} & FT(A^\omega \times (A \times \bar{Z} + \bar{Z})) & \xrightarrow{FT \phi} & FT(A^\omega \times \bar{Z}) \\
\downarrow & \searrow & \downarrow F \text{st}_{A^\omega, A \times \bar{Z} + \bar{Z}}^T & & \downarrow \\
A^\omega \times FT \bar{Z} & \xrightarrow{\text{st}_{A^\omega, \dots}^F} & F(A^\omega \times T(A \times \bar{Z} + \bar{Z})) & & \\
\downarrow & \searrow & \downarrow & & \downarrow \\
A^\omega \times FT \bar{Z} & \xrightarrow{\text{st}_{A^\omega, \bar{T} \bar{Z}}^F} & F(A^\omega \times \bar{T} \bar{Z}) & \xrightarrow{F \theta_{\bar{Z}}^\sharp} & FT(A^\omega \times \bar{Z})
\end{array}$$

where (i) commutes by (24) (note that $\Delta \times \text{id} = \text{st}_{A^\omega, A \times \bar{Z}, F\bar{Z}}^\times$), (ii) by (25) and by the naturality of the two inclusions, (iii) by definition of ϕ , (iv) by (25), (v) by naturality of ρ , (vi) by (24), (vii) by naturality of st^F and (viii) follows easily from the definition of θ and θ^\sharp , once it is noticed that st^T is the inductive extension of st^Σ . The outer shape of this diagram is (iv) of (33).

(v) in (33) commutes. First, chase the diagram

$$\begin{array}{c}
 \begin{array}{ccc}
 & A^\omega \times \bar{Z} & \xrightarrow{\text{ev}} & Z \\
 & \swarrow \text{id} \times \langle \text{id}, \bar{\zeta} \rangle & \downarrow \langle \text{id}, \text{id} \rangle & \\
 A^\omega \times \bar{Z} \times (F\bar{Z})^A & & A^\omega \times \bar{Z} \times A^\omega \times \bar{Z} & \\
 \downarrow \text{id} \times \eta_{\bar{Z}} \times \text{id} & \text{(i)} & \downarrow \text{id} \times \eta_{\bar{Z}} \times \text{id} \times \bar{\zeta} & \text{(iii)} \\
 A^\omega \times (A \times \bar{Z})^A \times (F\bar{Z})^A & & A^\omega \times (A \times \bar{Z})^A \times A^\omega \times (F\bar{Z})^A & \\
 \downarrow \text{id} \times \text{cs}_{A, A \times \bar{Z}, F\bar{Z}}^\times & \Delta \times \text{id} & \downarrow e_{A \times \bar{Z}} \times e_{F\bar{Z}} & \\
 A^\omega \times (A \times \bar{Z} \times F\bar{Z})^A & & A^\omega \times (A \times \bar{Z})^A \times A^\omega \times (F\bar{Z})^A & \\
 \downarrow e_{A \times \bar{Z} \times F\bar{Z}} & \text{(ii)} & \downarrow e_{A \times \bar{Z}} \times e_{F\bar{Z}} & \\
 A^\omega \times A \times \bar{Z} \times F\bar{Z} & & A^\omega \times A \times \bar{Z} \times A^\omega \times F\bar{Z} & \\
 \downarrow \Delta \times \text{id} & & \downarrow \langle \text{id}, \text{hd} \rangle^{-1} \times \text{id} \times \text{st}_{A^\omega, \bar{Z}}^F & \\
 & & A^\omega \times \bar{Z} \times F(A^\omega \times \bar{Z}) & \xrightarrow{\text{ev} \times F\text{ev}} & Z \times FZ \\
 & & & & \downarrow \langle \text{id}, \zeta \rangle
 \end{array}
 \end{array}$$

whose outer shape, mapped along Σ , is (v) of (33). Here, (i) is trivial and (ii) commutes by the definition (26) of e and by (22). The area (iii) is of the general shape

$$\begin{array}{ccc}
 X & \xrightarrow{f} & Y \\
 \langle \text{id}, \text{id} \rangle \downarrow & & \downarrow \langle g, h \rangle \\
 X \times X & \xrightarrow{k \times l} & U \times V
 \end{array}$$

(put $X = A^\omega \times \bar{Z}$, $Y = U = Z$, $V = FZ$), which commutes if and only if $k = g \circ f$ and

$l = h \circ f$, hence it is enough to check that the following two diagrams commute.

$$\begin{array}{ccc}
 A^\omega \times \bar{Z} & \xrightarrow{\text{ev}} & Z \\
 \text{id} \times \eta_{\bar{Z}} \downarrow & & \downarrow \text{id} \\
 A^\omega \times (A \times \bar{Z})^A & & \\
 e_{A \times \bar{Z}} \downarrow & & \\
 A^\omega \times A \times \bar{Z} & & \\
 \langle \text{tl}, \text{hd} \rangle^{-1} \downarrow & & \\
 A^\omega \times \bar{Z} & \xrightarrow{\text{ev}} & Z
 \end{array}
 \qquad
 \begin{array}{ccc}
 A^\omega \times \bar{Z} & \xrightarrow{\text{ev}} & Z \\
 \text{id} \times \bar{\zeta} \downarrow & & \downarrow \zeta \\
 A^\omega \times (F\bar{Z})^A & & \\
 e_{F\bar{Z}} \downarrow & & \\
 A^\omega \times F\bar{Z} & & \\
 \text{st}_{A^\omega, \bar{Z}}^F \downarrow & & \\
 F(A^\omega \times \bar{Z}) & \xrightarrow{F\text{ev}} & FZ
 \end{array}$$

But the diagram on the left commutes easily by the definition (26) of e , and the one on the right is (27). QED

Acknowledgments. The authors are grateful to an anonymous referee for thorough comments, and for spotting several mistakes in a previous version of this paper.

References

- Aceto, L., Fokkink, W. J., and Verhoef, C. (2001). Structural operational semantics. In Bergstra, J. A., Ponse, A., and Smolka, S., editors, *Handbook of Process Algebra*, pages 197–292. Elsevier.
- Aczel, P. and Mendler, N. (1989). A final coalgebra theorem. In Pitt, D., Rydeheard, D., Dybjer, P., Pitts, A., and Poigné, A., editors, *Category Theory and Computer Science*, volume 389 of *Lecture Notes in Computer Science*, pages 357–365.
- Bartels, F. (2004). *On Generalised Coinduction and Probabilistic Specification Formats*. PhD thesis, Vrije Universiteit Amsterdam.
- Carton, O. (2000). Wreath product and infinite words. *Journal of Pure and Applied Algebra*, 153:129–150.
- Hansen, H. and Klin, B. (2010). Pointwise extensions of GSOS-defined operations (extended abstract). In *CMCS 10 - Short Contributions*, CWI Technical Report SEN-1004, pages 10–11.
- Hansen, H. and Rutten, J. (2010). Symbolic synthesis of Mealy machines from arithmetic bitstream functions. Research Report SEN-1006, Centrum Wiskunde & Informatica (CWI), Amsterdam, The Netherlands.
- Hehner, E. and Horspool, R. (1979). A new representation of the rational numbers for fast easy arithmetic. *SIAM Journal on Computing*, 8:124–134.
- Jacobs, B. (2006). A bialgebraic review of deterministic automata, regular expressions and languages. In Futatsugi, K., Jouannaud, J.-P., and Meseguer, J., editors, *Algebra, Meaning and Computation: Essays dedicated to Joseph A. Goguen on the Occasion of his 65th Birthday*, volume 4060 of *Lecture Notes in Computer Science*, pages 375–404. Springer.

- Klin, B. (2009). Bialgebraic methods and modal logic in structural operational semantics. *Information and Computation*, 207:237–257.
- Koblitz, N. (1984). *P-adic numbers, p-adic Analysis, and Zeta-Functions*. Graduate Texts in Mathematics. Springer, 2nd edition.
- Lambek, J. (1968). A fixpoint theorem for complete categories. *Mathematische Zeitschrift*, 103(2):151–161.
- Lenisa, M., Power, J., and Watanabe, H. (2004). Category theory for operational semantics. *Theoretical Computer Science*, 327(1-2):135–154.
- Pin, J.-E. and Weil, P. (2002). The wreath product principle for ordered semigroups. *Communications in Algebra*, 30:5677–5713.
- Rutten, J. (1991). Coalgebra, concurrency and control. Research Report SEN-9921, Centrum Wiskunde & Informatica (CWI).
- Rutten, J. (2000). Universal coalgebra: a theory of systems. *Theoretical Computer Science*, 249:3–80.
- Rutten, J. (2003). Behavioural differential equations: a coinductive calculus of streams, automata and power series. *Theoretical Computer Science*, 308(1):1–53.
- Rutten, J. (2006). Algebraic specification and coalgebraic synthesis of Mealy machines. In *Proceedings of the 2nd International Workshop on Formal Aspects of Component Software (FACS 2005)*, volume 160 of *Electronic Notes in Theoretical Computer Science*, pages 305–319. Elsevier Science Publishers.
- Rutten, J. and Turi, D. (1994). Initial algebra and final coalgebra semantics for concurrency. In de Bakker, J., de Roever, W.-P., and Rozenberg, G., editors, *A Decade of Concurrency, Reflections and Perspectives, REX School/Symposium*, volume 803 of *Lecture Notes in Computer Science*, pages 530–582. Springer.
- Straubing, H. (1989). The wreath product and its applications. In *Formal Properties of Finite Automata and Applications*, volume 386 of *Lecture Notes in Computer Science*, pages 15–24. Springer.
- Turi, D. and Plotkin, G. (1997). Towards a mathematical operational semantics. In *Proceedings of the 12th Annual IEEE Symposium on Logic in Computer Science (LICS 1997)*, pages 280–291. IEEE Computer Society.
- Worrell, J. (2005). On the final sequence of a finitary set functor. *Theoretical Computer Science*, 338:184–199.