Coalgebra: Basic Concepts

Paul Blain Levy, University of Birmingham

April 12, 2022

1 Transition Systems

Integer Ltd. makes integer I/O machines, which have a button and a display. You press the button and it prints an integer. You press the button again and it prints another integer.

A machine has a variety of internal states. When you press the button, it's the current state that determines what integer gets printed, and what the new state will be (it could be a different state, or it could be the same state).

A machine is described by

- a set X (the set of states)
- a function $\zeta : X \longrightarrow \mathbb{Z} \times X$ (what happens when you press the button)
- the current state $x_0 \in X$

Exercise 1 *Machine number 392 has $\mathbb{Z} \times \mathbb{Z}$ as set of states. The behaviour function is $\zeta : \langle n, n' \rangle \mapsto \langle n + n', \langle n' + 1, n - 2 \rangle \rangle$. The current state is $\langle 4, 6 \rangle$. What is printed when you press the button three times?*

A rival company Integer And Boolean Inc. makes machines with three buttons and a display. If you press the red button or the green button it prints an integer, but if you press the bright pink button it prints a boolean. Such a machine is described by

- a set X (the set of states)
- a function $\zeta_{\mathsf{red}} : X \longrightarrow \mathbb{Z} \times X$ (what happens when you press the red button)
- a function $\zeta_{\mathsf{green}} : X \longrightarrow \mathbb{Z} \times X$ (what happens when you press the green button)
- a function $\zeta_{\mathsf{brightpink}} : X \longrightarrow \mathbb{B} \times X$ (what happens when you press the bright pink button).
- the current state $x_0 \in X$

Exercise 2 *Machine number 25 has $\mathbb{Z} \times \mathbb{Z}$ as set of states. The behaviour functions are*

$$\begin{aligned} \zeta_{\mathsf{red}} &: \langle n, n' \rangle \mapsto && \langle n, \langle n'+1, n-2 \rangle \rangle \\ \zeta_{\mathsf{green}} &: \langle n, n' \rangle \mapsto && \langle n'+1, \langle n+n', 2n' \rangle \rangle \\ \zeta_{\mathsf{brightpink}} &: \langle n, n' \rangle \mapsto && \langle n > n', \langle n', n' \rangle \rangle \end{aligned}$$

The current state is $\langle 3, 7 \rangle$. What is printed when you press the red button, then the green button, then the bright pink button, then the red button again?

Another company Interactive Integer make machines with a keyboard and a display. If you enter an integer, it prints another integer. Such a machine is described by

- a set X (the set of states)
- a function $\zeta : \mathbb{Z} \times X \longrightarrow \mathbb{Z} \times X$
- the current state $x_0 \in X$.

Exercise 3 *Machine number 40 has $\mathbb{Z} \times \mathbb{Z}$ as set of states. The behaviour function is given by*

$$\zeta : \langle m, \langle n, n' \rangle \rangle \mapsto \langle m+n, \langle 2m+n', n-1 \rangle \rangle$$

The current state is $\langle 4, 4 \rangle$. What is printed when you enter 5, then 3, then 5 again?

A somewhat unsuccessful company Unreliable Integer makes machines with a button and a display. If you press the button it might print an integer or it might print one of three error messages:
CRASH
BANG
WALLOP
Then the button jams shut and remains so forever. Such a machine is described by

- a set X (the set of states)
- a function $X \longrightarrow \mathbb{Z} \times X + E$, where E is the set of error messages,
- the current state $x_0 \in X$.

Exercise 4 *Machine number 6 has $\mathbb{Z} \times \mathbb{Z}$ as set of states. The behaviour function is described by*

$$\zeta : \langle n, n' \rangle \mapsto \begin{cases} \mathsf{inl}\ \langle n+3, \langle n', 7 \rangle \rangle & \textit{if } n' \leqslant 4 \\ \mathsf{inr}\ \text{BANG} & \textit{otherwise} \end{cases}$$

The current state is $\langle 3, 2 \rangle$. What is printed if you press the button twice?

A more popular company is Probabilistic Integer. If you press the button it consults some random data to decide what integer to print. The machine is described by

- a set X (the set of states)
- a function $\zeta : X \times (\mathbb{Z} \times X) \longrightarrow [0,1]$, where $\sum_{\langle n,y\rangle \in \mathbb{Z}\times X} \zeta(x, \langle n, y\rangle) = 1$ for each $x \in X$.
- the current state $x_0 \in X$.

A newcomer to the market is Nondeterministic Integer who make machines with a button and a display. If you press the button it prints an integer. But the behaviour doesn't just depend on the internal state, it also depends on a monkey hidden inside the machine. The machine is described by

- a set X (the set of states)
- a relation $r : X \relbar\joinrel\mapstochar\joinrel\rightarrow X$
- the current state $x_0 \in X$.

Exercise 5 *Machine number 24 has set of states $\mathbb{Z}\times\mathbb{Z}$. The behaviour relation is described by*

$$\langle n, n'\rangle \; r \; \langle m, \langle p, p'\rangle\rangle \overset{\text{def}}{\Leftrightarrow} m > n \textit{ and } p = p' + n$$

The current state $\langle 2, 5\rangle$ is. Describe one possible output if you press the button three times.

2 Coalgebras

These descriptions have more in common than appears at first sight. A machine consists of a set X together with a function

- $X \longrightarrow \mathbb{Z} \times X$ (Integer Ltd.)
- $X \longrightarrow (\mathbb{Z} \times X) \times (\mathbb{Z} \times X) \times (\mathbb{B} \times X)$ (Integer And Boolean Inc.)
- $X \longrightarrow (\mathbb{Z} \times X)^{\mathbb{Z}}$ (Interactive Integer)
- $X \longrightarrow \mathbb{Z} \times X + E$ (Unreliable Integer)
- $X \longrightarrow D(\mathbb{Z} \times X)$ (Probabilistic Integer), where DY is the set of discrete probability distributions on Y.
- $X \longrightarrow \mathcal{P}(\mathbb{Z} \times X)$ (Nondeterministic Integer)

and a current state $x_0 \in X$.

Definition 6 *Let $\mathcal{C}$ and $\mathcal{D}$ be categories. A* functor $F : \mathcal{C} \longrightarrow \mathcal{D}$ *associates*

- *to each $\mathcal{C}$-object X, a $\mathcal{D}$-object FX*
- *to each $\mathcal{C}$-morphism* $X \xrightarrow{f} Y$ *, a $\mathcal{D}$-morphism* $FX \xrightarrow{Ff} FY$

in such a way that

- *for every object X we have $F\mathsf{id}_X = \mathsf{id}_{FX}$*
- *for any morphisms* $X \xrightarrow{f} Y \xrightarrow{g} Z$ *we have $F(f;g) = Ff;Fg$.*

A *endofunctor* on a category $\mathcal{C}$ is a functor $F : \mathcal{C} \longrightarrow \mathcal{C}$.
For example, there's an endofunctor on **Set** that sends

- a set X to the set $\mathbb{Z} \times X$
- a function $X \xrightarrow{f} Y$ is mapped to the function $\mathbb{Z} \times X \xrightarrow{\mathbb{Z}\times f} \mathbb{Z} \times Y$ that sends $\langle n, x\rangle$ to $\langle n, f(x)\rangle$.

Typically we write a functor by saying only what it does to objects, but this is sloppy.

Definition 7 *Let $\mathcal{C}$ be a category and let F be an endofunctor on $\mathcal{C}$. An F-*coalgebra *consists of*

- *a $\mathcal{C}$-object X, the* carrier
- *a $\mathcal{C}$-morphism $\zeta : X \longrightarrow FX$, the* structure*.*

We call X the carrier *of the coalgebra and ζ the* structure *of the coalgebra.*

For example, a machine made by Integer Ltd. is a $X \mapsto \mathbb{Z} \times X$ coalgebra. Only one thing is missing: a coalgebra does not have a current state. If F is an endofunctor on **Set**, we say that a *pointed* F-coalgebra is an F-coalgebra (X, ζ) together with a state $x_0 \in X$. In general a *pointed* set is a set X together with an element $x_0 \in X$.

What about the other machines? Each of these is given as a (pointed) coalgebra for a suitable endofunctor on **Set**.

- If F, G, H are endofunctors on **Set** then so is $X \mapsto FX \times GX \times HX$, with $X \xrightarrow{f} Y$ mapping to

$$FX \times GX \times HX \xrightarrow{Ff\times Gf\times Hf} FY \times GY \times HY$$

  that sends (a, b, c) to $((Ff)a, (Gf)b, (Hf)c)$, and so is $X \mapsto FX + GX + HX$.

- $X \mapsto X^{\mathbb{Z}}$ is an endofunctor, with $X \xrightarrow{f} Y$ mapping to

$$X^{\mathbb{Z}} \xrightarrow{f^{\mathbb{Z}}} Y^{\mathbb{Z}}$$

  that sends $(a_i)_{i\in I}$ to $(f(a_i))_{i\in\mathbb{Z}}$.

- $X \mapsto X + E$ is an endofunctor, with $X \xrightarrow{f} Y$ mapping to
$$X + E \xrightarrow{f+E} Y + E$$
that sends $\mathsf{inl}\ x$ to $\mathsf{inl}\ f(x)$ and $\mathsf{inr}\ e$ to $\mathsf{inr}\ e$.

- The endofunctor D maps X to the set of discrete distributions on X is an endofunctor. A *discrete distribution* is a function $d : X \longrightarrow [0, 1]$ such that $\sum_{x \in x} d(x) = 1$. The function $X \xrightarrow{f} Y$ is mapped to $DX \xrightarrow{Df} DY$ that sends d to $y \mapsto \sum_{x \in f^{-1}(y)} d(x)$.

- The endofunctor $\mathcal{P}$ maps X to the set of subsets of X. A function $X \xrightarrow{f} Y$ is mapped to $\mathcal{P}X \xrightarrow{\mathcal{P}f} \mathcal{P}Y$ that sends U to $\{f(x) \mid x \in U\}$.

Exercise 8 *Accepting Integer makes machines that consist of*

- *a set X of states*
- *a function $\zeta : X \longrightarrow \mathbb{Z} \times X$*
- *a subset $U \subseteq X$ of* accepting states
- *a current state $x_0 \in X$*

What endofunctor on **Set** *is such a machine a pointed coalgebra for?*

3 Subfunctors

Let F be an endofunctor on **Set**. A *subfunctor G* of F associates to each set X a subset GX of FX, in such a way that for any function $X \xrightarrow{f} Y$ and element $a \in GX$, we have $(Ff)a \in GY$. This enables us to define $GX \xrightarrow{Gf} GY$ to be Ff, so G is also an endofunctor on **Set**. If we have an F-coalgebra (X, ζ) we can ask: is it a G-coalgebra? In other words, is $\zeta(x) \in GX$ for all $x \in X$?

For example, $D_{\mathsf{fin}}X$ is the set of *finite distributions* on X, i.e. those $d \in DX$ such that the set $\{x \in X \mid d(x) > 0\}$ is finite. A D_{fin}-coalgebra is a special kind of probabilistic transition system.

Exercise 9 *Which of these are subfunctors of $\mathcal{P}$? A set X is sent to:*

- *The set of inhabited subsets of X. (Hint: yes)*
- *The set of finite subsets of X. (Hint: yes)*
- *The set of subsets of X of size at most* 3*.*
- *The set of finite subsets of even size.*
- *The set of countable subsets of X.*

(If you know about cardinals:) Give all the subfunctors of $\mathcal{P}$.

Thus we have *lively* transition systems and *finitely branching* transition systems.

4 Active and Passive States

In the examples above, the states of the system are *passive*, waiting for input from outside. We could also consider a set of *active* states, that are executing a program and will then output. For example, a machine made by Interactive Input could be described as

- a set Y of active states
- a function $\xi : Y \longrightarrow \mathbb{Z} \times (Y^{\mathbb{Z}})$
- a current state $y_0 \in Y$.

Or it could be described as

- a set X of passive states
- a set Y of active states
- a function $\zeta : X \longrightarrow Y^{\mathbb{Z}}$
- a function $\xi : Y \longrightarrow \mathbb{Z} \times X$.

together with a passive state $x_0 \in X$ or passive state $y_0 \in Y$.

Each of these (leaving aside the current state) is a coalgebra. In the last case we use an endofunctor on $\mathbf{Set}^2$ that maps (X, Y) to $(Y^{\mathbb{Z}}, \mathbb{Z} \times X)$.

5 The Category of Coalgebras

Of course we want to make coalgebras into a category.

Definition 10 *Let $\mathcal{C}$ be a category and let F be an endofunctor on $\mathcal{C}$. Let (X, ζ) and (Y, ξ) be F-coalgebras. A* F-coalgebra morphism *From (X, ζ) to (Y, ξ) is a $\mathcal{C}$-morphism $X \xrightarrow{f} Y$ such that*

$$\begin{tikzcd} FX \arrow[r, "Ff"] & FY \\ X \arrow[u, "\zeta"] \arrow[r, "f"'] & Y \arrow[u, "\xi"'] \end{tikzcd} \qquad \textit{commutes.}$$

Now we get a category $\mathbf{Coalg}(F)$ whose objects are F-coalgebras and whose morphisms are F-coalgebra morphisms. Composition and identities are the same as in $\mathcal{C}$.

6 The Category of Algebras

The dual notion is that of F*-algebra*, which consists of

- a $\mathcal{C}$-object X, the *carrier*
- a $\mathcal{C}$-morphism $\zeta : FX \longrightarrow X$.

An F*-coalgebra morphism* From (X,θ) to (Y,ϕ) is a $\mathcal{C}$-morphism $X \xrightarrow{h} Y$ such that

$$\begin{tikzcd} FX \arrow[r, "Fh"] \arrow[d, "\theta"'] & FY \arrow[d, "\phi"] \\ X \arrow[r, "h"'] & Y \end{tikzcd}$$

commutes.

So we get a category $\mathbf{Alg}(F)$ whose objects are F-algebras and whose morphisms are F-algebra morphisms. Composition and identities are the same as in $\mathcal{C}$.

Here's an example. A *pointed magma* $(X, e, *)$ consists of a set X with an element e and binary operation $*$. (It's a *monoid* when the associativity law, right unital law and left unital law are satisfied.) A *homomorphism* from $(X, e, *)$ to $(Y, p, \otimes)$ is a function $h : X \to Y$ that preserves the point and the binary operation. The category of pointed magmas and homomorphisms is (isomorphic to) the category of F-algebras, for a suitable endofunctor F on **Set**. Which?

7 Coalgebra-to-algebra morphisms

A *morphism* from an F-coalgebra (X,ζ) to an F-algebra (Y,ϕ) is a $\mathcal{C}$-morphism $X \xrightarrow{g} Y$ such that

$$\begin{tikzcd} FX \arrow[r, "Fg"] & FY \arrow[d, "\phi"] \\ X \arrow[u, "\zeta"] \arrow[r, "Fg"'] & Y \end{tikzcd}$$

commutes.

So we get a *bimodule* from the category $\mathbf{Coalg}(F)$ to the category $\mathbf{Alg}(F)$. That means that we can compose a coalgebra-to-algebra morphism g with a coalgebra morphism f on the left, or a coalgebra morphism h on the right, and the associativity laws for $f'; f; g$ and $g; h; h'$ and $f; g; h$ are all satisfied.

Exercise 11 *Let F be the endofunctor on* **Set** *sending a set X to $1 + \mathbb{Z} + X^2$, which we write as*

$$\{\mathsf{Nil}\} \cup \{\mathsf{Just}\, n \mid n \in \mathbb{Z}\} \cup \{\mathsf{Parts}(x, y) \mid x, y \in X\}$$

Let A be the set of lists of integers and B the set of sorted lists of integers. (Repetitions are allowed.) Let $\zeta : A \to FA$ be the function sending

- *the empty list to* Nil
- *the list* $[n]$ *to* $\mathsf{Just}\, n$
- *a list* $s +\!\!+ t$*, where* $|s| > 0$ *and* $|t| \in \{|s|, |s|+1\}$*, to* $\mathsf{Parts}\,(s,t)$*.*

Let $\phi : FB \to B$ *be the function sending*

- Nil *to the empty list*
- $\mathsf{Just}\, n$ *to* $[n]$
- $\mathsf{Parts}(s,t)$ *to the merge of* s *and* t*.*

Show that there is a unique morphism from the F*-coalgebra* (A,ζ) *to the* F*-algebra* (B,ϕ)*.*

8 Final Coalgebras

Definition 12 *In a category* $\mathcal{C}$*, an* isomorphism *is a morphism* $f : x \to y$ *such that there's a (necessarily unique) morphism* $g : y \to x$ *such that*

$$\begin{tikzcd}
x \arrow[r, "f"] \arrow[dr, "\mathsf{id}"'] & y \arrow[d, "g"] \arrow[dr, "\mathsf{id}"] & \\
& x \arrow[r, "f"'] & y
\end{tikzcd}$$

For example, the isomorphisms in **Set** are the bijections.

Definition 13 *Let* $\mathcal{C}$ *be a category. An object* x *is*

1. initial *if for every object* y *there's a unique morphism* $x \to y$*.*
2. terminal *(or* final*) if for every object* y *there is a unique morphism* $y \to x$*.*

Note that the initial object is unique up to unique isomorphism, and likewise the terminal object.

A map from a (specified) initial algebra to an algebra (X,θ) is called a *catamorphism.* A map from a coalgebra (Y,ζ) to a (specified) final coalgebra is called a *anamorphism.*

Lemma 14 *(Lambek's lemma) Let* $\mathcal{C}$ *be a category with endofunctor* F*.*

- *Let* (X,θ) *be an initial* F*-algebra. Then* θ *is an isomorphism.*
- *Same for final coalgebra.*

A *final* F*-coalgebra* N is a final object in the category $\mathbf{Coalg}(F)$. Thus from any coalgebra M there is a unique coalgebra morphism from M to N. It is called the *anamorphism.*

Lemma 15 *(Lambek's Lemma) Let F be an endofunctor on a category $\mathcal{C}$. Let (X, ζ) be a final coalgebra. Then $X \xrightarrow{\zeta} FX$ is an isomorphism.*

(Note: for a set A, we write A^* for the set of lists of elements of A, and A^ω for the set of infinite sequences of elements of A.)

If you buy a machine from Unreliable Integer, i.e. a pointed coalgebra for $X \mapsto E + \mathbb{Z} \times X$, its full behaviour over time is described by a finite list of integers followed by an error, of an infinite list of integers. Two machines with the same infinite trace are *trace equivalent*. They are equivalent for all practical purposes. Admittedly they have different states, but those states are internal so you cannot observe them.

So why bother with states at all? An employee at Unreliable Integer makes a machine in which the set of states is $\mathbb{Z}^* \times E + \mathbb{Z}^\omega$, the set of behaviours. Then the behaviour of a state s is actually s.

Theorem 16 *Let A and B be sets. The endofunctor on* **Set** *sending X to $B + A \times X$ has final coalgebra $B \times A^* + A^\omega$ with structure*

$$\begin{array}{rcl} A^* \times B + A^\omega & \cong & (1 + A \times A^*) \times B + A \times A^\omega \\ & \cong & B + A \times (A^* \times B + A^\omega) \end{array}$$

and initial algebra $B \times A^*$ with inverse structure

$$\begin{array}{rcl} A^* \times B & \cong & (1 + A \times A^*) \times B \\ & \cong & B + A \times (A^* \times B) \end{array}$$

If you buy a machine form Interactive Integer, i.e. a pointed coalgebra for $X \mapsto \mathbb{Z} \to (\mathbb{Z} \times X)$, then a behaviour converts a nonempty list of integers (input) into an integer (output).

Theorem 17 *The endofunctor on* **Set** *sending X to $A \to (B \times X)$ has final coalgebra $A^* \to B$ with structure*

$$\begin{array}{rcl} A^* \to B & \cong & (1 + A \times A^*) \to B \\ & \cong & B \times (A \to (A^* \to B)) \end{array}$$

Theorem 18 *The endofunctor on* **Set** *sending X to $B \times (A \to X)$ has final coalgebra $A^* \to B$ with structure*

$$\begin{array}{rcl} A^* \to B & \cong & (1 + A \times A^*) \to B \\ & \cong & B \times (A \to (A^* \to B)) \end{array}$$

9 Infinite Trees

If you buy a machine from Integer and Boolean Inc., the full behaviour is defined by an infinite tree rather than an infinite list. To be more precise, consider *finite traces* such as the following:

I pressed the red button.
The machine printed 17.
I pressed the bright pink button.
The machine printed TRUE.
I pressed the red button.
The machine printed 42.

A finite trace is a sequence $a_0, b_0, a_1, b_1, \ldots, a_{n-1}, b_{n-1}$ where each a_i is a button and b_i is an appropriate response (integer if a_i is the red button or green button, boolean if a_i is the bright pink button).

Now an infinite tree is a set U of finite traces with the following properties:

- the *empty trace* $\varepsilon \in U$
- if s and t are traces and s is a *prefix* of t and $t \in U$ then $s \in U$.
- if $s \in U$ and a is a button then there is a unique appropriate response b to a such that $s + (a, b) \in U$.

Now if U is an infinite tree, then for each button a

- let b_a be the response such that $(a, b_a) \in U$
- let U_a be the set of all traces t such that $(a, b_a) + t \in U$.

The set of infinite trees, with the function ζ mapping U at a to (b_a, U_a), forms a coalgebra for

$$X \mapsto (\mathbb{Z} \times X) \times (\mathbb{Z} \times X) \times (\mathbb{B} \times X)$$

This is a final coalgebra.

10 The Rolling Rule

Let $F : \mathcal{C} \longrightarrow \mathcal{D}$ and $G : \mathcal{D} \longrightarrow \mathcal{D}$ be functors. If (x, θ) is an initial GF-algebra, then $(Fx, F\theta)$ is an initial FG-algebra.

For example, once we know a final coalgebra for the endofunctor $B \times (A \rightarrow -)$, we can obtain a final coalgebra for the endofunctor $A \rightarrow (B \times -)$.

11 Limits of algebras and colimits of coalgebras

Let F be an endofunctor on $\mathcal{C}$. If $\mathcal{C}$ has all limits then $\mathbf{Alg}(F)$ does too. If $\mathcal{C}$ has all colimits then $\mathbf{Coalg}(F)$ does too.

We can say more: the forgetful functor $U : \mathbf{Alg}(F) \rightarrow \mathcal{C}$ *creates limits*, and the forgetful functor $\mathbf{Coalg}(F) \rightarrow \mathcal{C}$ *creates coimits.*

This means that if we have a diagram $D : \mathbb{I} \rightarrow \mathbf{Alg}(F)$, which gives a diagram $UD : \mathbb{I} \rightarrow \mathcal{C}$, and this has a limit $(V, (\mathsf{p_i})_{i \in \mathbb{I}})$, then there's a unique algebra structure $\theta : FV \rightarrow V$ making all the projections $(\mathsf{p_i})_{i \in \mathbb{I}}$ into algebra morphisms, and the resulting cone is a limit.

12 Inductive and coinductive definition of predicates

Let A and B be posets. A function $f : A \to B$ is *monotone* when $a \leqslant b$ implies $f(a) \leqslant f(b)$.

Let A be a poset and f a monotone endofunction on A. An element $a \in A$ is a *prefixpoint* of f when $f(x) \leqslant x$, and a postfixpoint of f when $x \leqslant f(x)$. We can look for a least prefixpoint (analogous to initial algebra) and for a greatest postfixpoint (analogous to final coalgebra). Each of these is a fixpoint.

Any infimum of prefixpoints is a prefixpoint, and any supremum of postfixpoints is a postfixpoint. If A is a complete lattice, then there must be a least prefixpoint, viz. the infimum of all prefixpoints. And likewise there must be a greatest postfixpoint.

Let's see an example. Let F be the endofunction on $\mathcal{P}\mathbb{N}$ that sends R to

$$\{7\} \cup \{0|1 \in R\} \cup \{1|0 \in R\} \cup \{m + n + 17 \mid m, n \in R\}$$

A subset R is a prefixpoint when

- $7 \in R$
- if $1 \in R$ then $0 \in R$
- if $0 \in R$ then $1 \in R$
- if $m, n \in R$ then $m + n + 17 \in R$.

So 7 and 31 are in the least prefixpoint, and everything in the least prefixpoint is $\geqslant 7$. However, the greatest postfixpoint includes 0 and 1.

13 Recursive coalgebras and corecursive algebra

An F-coalgebra is *recursive* when there's a unique map from it to every F-algebra.

For the functors we've seen (preserve monos and inverse images), corecursive coincides with *well-founded*, i.e. no element has an infinite trace. (Due to Paul Taylor.)

An F-algebra is *corecursive* when there's a unique map from every F-coalgebra to it.

Applying F preserves these properties. Limit of algebras and colimit of coalgebras preserve these properties.

There are other constructions that generate recursive coalgebras and corecursive algebras. They're called *recursion principles*.

14 The inductive and coinductive chain

Let f be an endofunction a poset A with suprema of well-ordered chains. Then we can form the least prefixpoint as follows. Form the increasing sequence $(c_\alpha)_{\alpha \in A}$ of postfixpoints that are $\leqslant$ every prefixpoint as follows:

- a_0 is the least element
- a_{n+1} is $f(a_n)$
- $a_n = \bigvee_{m<n} a_m$ if n is a limit.

If a_n is a fixpoint, then it's the least prefixpoint. Conversely, if f has a least prefixpoint, it's achieved at some n.

What about for initial algebras? If F is an endofunctor on a category $\mathcal{C}$ with colimits of ordinal chains, we can form the inductive chain that consists of recursive coalgebras.

$$\begin{tikzcd} a_0 \arrow[r, "f_0"] & a_1 \arrow[r, "f_1"] & a_2 \arrow[r, "f_2"] & \cdots \end{tikzcd}$$

If f_n is an isomorphism, we have an initial F-algebra. (This is called Adámek's theorem.)

Exercise 19 *Show that a polynomial functor preserves the limit of every (inhabited) connected diagram. Deduce that the final coalgebra is reached at ω.*

15 Bisimulation

I've bought two machines from Integer and Boolean Inc.

Machine I has state set $X = \{A, B\}$. Pressing the red button

- from state A, prints 3 and remains in state A
- from state B, prints 5 and moves to state A

Pressing the green button

- from state A, prints 8 and moves to state B
- from state B, prints 4 and remains in state B

Pressing the bright pink button

- from state A, prints TRUE and remains in state A
- from state B, prints FALSE and moves to state A.

The current state is $x_0 = A$.

Machine II has state set $X' = \mathbb{N}$. Pressing the red button

- from state $n < 6$, prints 4 and moves to state $n + 7$

- from even state $n \geqslant 6$, prints 3 and moves to state $n+2$
- from odd state $n \geqslant 6$, prints 5 and moves to state $n+5$

Pressing the green button

- from state $n < 6$, prints 9 and remains in state n
- from even state $n \geqslant 6$, prints 8 and moves to state $n+25$
- from odd state $n \geqslant 6$, prints 4 and remains in state n

Pressing the bright pink button

- from state $n < 6$, prints FALSE and moves to state $n+1$
- from even state $n \geqslant 6$, prints TRUE and moves to state $n+8$
- from odd state $n \geqslant 6$, prints FALSE and moves to state $n+13$

The current state is $x'_0 = 10$.

I want to show these two machines have the same anamorphic image—set of finite traces. But actually writing out the set of finite traces is difficult. There is an alternative method.

Suppose that $\mathcal{R}$ is a relation from X to X', with the following property. For any $x \mathrel{\mathcal{R}} x'$ and button a, we have $\zeta_a x = \langle n, y \rangle$ and $\zeta_a x' = \langle n, y' \rangle$ with $y \mathrel{\mathcal{R}} y'$. Thus

> related states
> applied to the same input
> give the same output
> ending up in related states.

Such a relation is called a *bisimulation* between the two transition systems.

Two pointed coalgebras (X, ζ, x_0) and (X', ζ', x'_0) are *bisimilar* when there is some bisimulation $\mathcal{R}$ from (X, ζ) to (X', ζ') such that $x_0 \mathrel{\mathcal{R}} x_1$.

For our example we could take $\mathcal{R}$ to be

$$\{(A, n) \mid n \geqslant 6, n \text{ even }\} \cup \{(B, n) \mid n \geqslant 6, n \text{ odd }\}$$

Theorem 20 *Let (X, ζ, x_0) and (X', ζ', x'_0) be pointed coalgebras for the endofunctor*

$$X \mapsto (\mathbb{Z} \times X) \times (\mathbb{Z} \times X) \times (\mathbb{B} \times X)$$

They are bisimilar iff they have the same anamorphic image (set of finite traces).

Exercise 21 *Machine III and Machine IV are produced by Interactive Integer.*

Machine III has set of states $X = \mathbb{Z}$.

- *A state $n > 0$, when it receives an input m, prints $m + n$ and moves to state $-m - 2n$.*

- *The state* 0*, when it receives an input* m*, prints* 17 *and moves to state* 2.
- *A state* $n < 0$*, when it receives an input* m*, prints* $m - n$ *and moves to state* $m - 2n$

The current state is $x_0 = 5$

Machine IV has set of states $\mathbb{Z} \times \mathbb{Z}$*. State* $\langle n, n' \rangle$*, receiving input* m*, prints* $m+n$ *and moves to state* $\langle m + 2n, m + n + n' \rangle$*. The current state is* $x'_0 = \langle 5, 7 \rangle$.

Show the two machines are bisimilar.

16 Nondeterminism

A machine made by Nondeterministic Integer is a pointed coalgebra for

$$X \mapsto \mathcal{P}(\mathbb{Z} \times X)$$

A machine made by Nondeterministic Integer And Boolean is a pointed coalgebra for

$$X \mapsto \mathcal{P}(\mathbb{Z} \times X) \times \mathcal{P}(\mathbb{Z} \times X) \times \mathcal{P}(\mathbb{B} \times X)$$

A machine made by Nondeterministic Interactive Integer is a pointed coalgebra for

$$X \mapsto (\mathcal{P}(\mathbb{Z} \times X))^{\mathbb{Z}}$$

Let (X, ζ) and (X', ζ') be coalgebras. Suppose that $\mathcal{R}$ is a relation from X to X' with the following property. For any $x \mathrel{\mathcal{R}} x'$ and input m,

- if $\langle n, y \rangle \in \zeta(x)m$ then $\langle n, y' \rangle \in \zeta'(x')m$ for some y' such that $y \mathrel{\mathcal{R}} y'$
- if $\langle n, y' \rangle \in \zeta'(x')m$ then $\langle n, y \rangle \in \zeta(x)m$ for some y such that $y \mathrel{\mathcal{R}} y'$

Then $\mathcal{R}$ is a *bisimulation.* If it has the first property, it's a *simulation.*

The largest bisimulation (i.e. the union of all bisimulations) from (X, ζ) to (X', ζ') is called *bisimilarity.* The largest simulation is called *similarity.*

These are coinductive definitions.

- Similarity is a preorder.
- Bisimilarity is an equivalence relation.
- Bisimilarity implies mutual similarity.
- Similarity implies finite and infinite trace inclusion. That means: if x is similar to y, then every finite or infinite trace of x is a finite or infinite trace of y.

Coalgebra morphisms are functional bisimulations

Due to Aczel, Mendler, Rutten, ...

Let (X, ζ) and (Y, ξ) be coalgebras.

Let $X \xrightarrow{f} Y$ be a function.

Then f is a coalgebra morphism iff f, regarded as a relation, is a bisimulation.

Corollary If $(X, \zeta) \xrightarrow{f} (Y, \xi)$ is a coalgebra morphism, then every $x \in X$ is bisimilar to $f(x)$.

Encompassment

(X, ζ) is *encompassed* by (Y, ξ) when for every state in X there is a bisimilar state in Y.

You can think of this as saying that (Y, ξ) is at least as expressive as (X, ζ).

This is a preorder on transition systems.

If $(X, \zeta) \xrightarrow{f} (Y, \xi)$ is a coalgebra morphism, then (X, ζ) is encompassed by (Y, ξ).

If f is a surjective coalgebra morphism, then (X, ζ) and (Y, ξ) are mutually encompassed.

17 Strongly Extensional Coalgebras

A coalgebra $M = (X, \zeta)$ is *extensional* when ζ is injective. It's *strongly extensional* when any two bisimilar states $x, x' \in X$ are equal. Such a coalgebra has various significant properties.

- Any coalgebra morphism from M is injective.
- Given another coalgebra N encompassed by M, there's a unique coalgebra morphism $N \xrightarrow{f} M$. It's the bisimilarity relation from N to M. Moreover N is strongly extensional iff f is injective.

Strongly extensional Quotients

Let $M = (X, \zeta)$ be a coalgebra. Let Y be X quotiented by bisimilarity. There's a unique $Y \xrightarrow{\xi} FY$ such that the quotient map $X \xrightarrow{p} Y$ is a coalgebra morphism from (X, ζ) to (Y, ξ). Moreover (Y, ξ) is strongly extensional.

Bisimilarity via Cospans

Two pointed coalgebras (X, ζ, x_0) and (Y, ξ, y_0) are bisimilar iff there is a *cospan* of coalgebra morphisms

$$\begin{tikzcd}
(X,\zeta) \arrow[dr, "f"] & & (Y,\xi) \arrow[dl, "g"'] \\
& (Z,\phi) &
\end{tikzcd}$$

such that $f(x_0) = g(y_0)$.

Final Coalgebras

Suppose $M = (X, \zeta)$ is an F-coalgebra. Then it is final iff it is all-encompassing and extensional.

Suppose $M = (X, \zeta)$ is a final F-coalgebra. Then two pointed F-coalgebras are bisimilar iff they have the same anamorphic image.

Suppose $M = (X, \zeta)$ is an F-coalgebra. Then it is all-encompassing iff its extensional quotient is final.

18 Finding An All-Encompassing Coalgebra

In the case of $\mathcal{P}$, there is no all-encompassing coalgebra.

But let's consider finitely nondeterministic or countably nondeterministic systems (X, ζ). Any state x has a countable set of descendants, and we can restrict ζ to this set to get a countable coalgebra. This is isomorphic to a coalgebra carried by a subset of $\mathbb{N}$.

Now take the sum of all coalgebras carried by a subset of $\mathbb{N}$. This is an all-encompassing system. So its extensional quotient is a final coalgebra.

19 Relators

Let F be an endofunctor on **Set**. An F*-relator* maps each relation $\begin{tikzcd}X \arrow[r, "\mathcal{R}", "\shortmid"{marking}] & Y\end{tikzcd}$ to a relation $\begin{tikzcd}FX \arrow[r, "\Gamma\mathcal{R}", "\shortmid"{marking}] & FY\end{tikzcd}$ in such a way that the following hold.

- For any relations $\begin{tikzcd}X \arrow[r, "{\mathcal{R},\mathcal{S}}", "\shortmid"{marking}] & Y\end{tikzcd}$, if $\mathcal{R} \subseteq \mathcal{S}$ then $\Gamma\mathcal{R} \subseteq \Gamma\mathcal{S}$.
- For any set X we have $(=_{FX}) \subseteq \Gamma(=_X)$
- For any relations $\begin{tikzcd}X \arrow[r, "\mathcal{R}", "\shortmid"{marking}] & Y \arrow[r, "\mathcal{S}", "\shortmid"{marking}] & Z\end{tikzcd}$ we have $(\Gamma\mathcal{R}); (\Gamma\mathcal{S}) \subseteq \Gamma(\mathcal{R}; \mathcal{S})$
- For any functions $Z \xrightarrow{f} X$ and $W \xrightarrow{g} Y$, and any relation $\begin{tikzcd}X \arrow[r, "\mathcal{R}", "\shortmid"{marking}] & Y\end{tikzcd}$, we have $\Gamma(f \times g)^{-1}\mathcal{R} = (Ff \times Fg)^{-1}\Gamma\mathcal{R}$.

Γ is a *conversive* relator when $\Gamma(\mathcal{R}^{\mathsf{c}}) = (\Gamma\mathcal{R})^{\mathsf{c}}$ for every relation $X \overset{\mathcal{R}}{\relbar\joinrel\mapstochar\joinrel\rightarrow} Y$.

Let (X, ζ) and (X', ζ') be F-coalgebras. Let Γ be an F-relator.

A relation $X \overset{\mathcal{R}}{\relbar\joinrel\mapstochar\joinrel\rightarrow} X'$ is a Γ-*simulation* when $x \, \mathcal{R} \, x'$ implies that $\zeta(x) \; \Gamma\mathcal{R} \; \zeta'(x')$.

By choosing different relators Γ, we get different notions of simulation and bisimulation.

Deterministic Examples

$\mathbb{Z} \times \mathcal{R}$ relates $\langle n, x \rangle$ to $\langle n, x' \rangle$ when $x \; \mathcal{R} \; x'$. This gives an $\mathbb{Z} \times -$ relator, the in-house relator of Integer Ltd.

$\mathcal{R} \times \mathcal{S} \times \mathcal{T}$ relates $\langle x, y, z \rangle$ to $\langle x', y', z' \rangle$ when $x \; \mathcal{R} \; x'$ and $y \; \mathcal{S} \; y'$ and $z \; \mathcal{T} \; z'$. We get the in-house relator of Integer and Boolean Inc.

$\mathcal{R}^{\mathbb{Z}}$ relates p to p' when $pm \; \mathcal{R} \; p'm$ for each input m. This gives the in-house relator of Interactive Integer.

$\mathcal{R} + E$ relates $\mathsf{inl}\ x$ to $\mathsf{inl}\ x'$ when $x \; \mathcal{R} \; x'$ and also relates $\mathsf{inr}\ e$ to $\mathsf{inr}\ e$. This gives the in-house relator of Unreliable Integer.

Nondeterministic Examples

We have two $\mathcal{P}$-relators.

$\mathsf{Sim}\ \mathcal{R}$ relates $U \in \mathcal{P}X$ to $V \in \mathcal{P}Y$ when

- for all $x \in U$ there exists $y \in V$ such that $x \; \mathcal{R} \; y$

This gives simulation.

$\mathsf{Bisim}\ \mathcal{R}$ relates $U \in \mathcal{P}X$ to $V \in \mathcal{P}Y$ when

- for all $x \in U$ there exists $y \in V$ such that $x \; \mathcal{R} \; y$
- for all $y \in V$ there exists $x \in U$ such that $x \; \mathcal{R} \; y$.

This gives bisimulation.

If G is a subfunctor of F, then any F-relator is also a G-relator.

Systems with Divergence

A system *diverges* (or *hangs*) when it runs forever without producing any output. For example, a machine made by Interactive Divergent Integer is a pointed coalgebra for

$$X \mapsto \mathcal{P}(\mathbb{Z} \times X + \{\Uparrow\})^{\mathbb{Z}}$$

This is similar to the Unreliable Integer machines we considered previously. Let (X, ζ) and (X', ζ') be such coalgebras. Let $X \overset{\mathcal{R}}{\relbar\joinrel\mapstochar\joinrel\rightarrow} X'$ be a relation.

$\mathcal{R}$ is an *inclusion simulation* when for any $x \; \mathcal{R} \; x'$ and input m,

- if $x \overset{m,n}{\rightsquigarrow} y$ then there exists y' such that $x' \overset{m,n}{\rightsquigarrow} y'$ and $x' \; \mathcal{R} \; y'$.
- if $xm \Uparrow$ then $x'm \Uparrow$.

If we just have the first condition, $\mathcal{R}$ is a *lower simulation.*

$\mathcal{R}$ is an *smash simulation* when for any $x\mathcal{R}\ x'$ and input m, if $xm \not\Uparrow$ then

- $x'm \not\Uparrow$
- if $x \overset{m,n}{\rightsquigarrow} y$ then there exists y' such that $x' \overset{m,n}{\rightsquigarrow} y'$ and $y\ \mathcal{R}\ y'$
- if $x' \overset{m,n}{\rightsquigarrow} y'$ then there exists y such that $x \overset{m,n}{\rightsquigarrow} y$ and $y\ \mathcal{R}\ y'$

If we just have the first and third conditions, $\mathcal{R}$ is an *upper simulation.*

If $\mathcal{R}$ is an upper and lower simulation, it's a *convex simulation.*

If $\mathcal{R}$ and its converse are a lower (resp. upper, convex) simulation, then $\mathcal{R}$ is a lower (resp. upper, convex) bisimulation

Altogether we obtain numerous (in fact nineteen) different relators on

$$X \mapsto \mathcal{P}(X + \{\Uparrow\})$$

Three of them are conversive.

Probabilistic Systems

DX is the set of (discrete) distributions on X.

We need a D-relator for bisimulation.

Given a relation $X \overset{\mathcal{R}}{\relbar\joinrel\mapstochar\joinrel\rightarrow} Y$, we defines a relation $DX \overset{D\mathcal{R}}{\relbar\joinrel\mapstochar\joinrel\rightarrow} DY$. This relates $d \in DX$ to $d' \in DY$ when

$$dU \leqslant d'\mathcal{R}(U)$$

for every $U \subseteq X$. Here

$$\mathcal{R}(U) \stackrel{\text{def}}{=} \{y \in Y \mid \exists x \in U.\ x\mathcal{R}y\}$$

This is a conversive relator.

Endofunctor on Preord

Preord is the category of preordered sets and monotone functions.

Our endofunctor F on **Set** lifts to an endofunction F_Γ on **Preord**.

- A preordered set $(X, \leqslant)$ maps to $(FX, \Gamma(\leqslant))$.
- A monotone function $A \xrightarrow{f} B$ maps to Ff.

Saying it's a *lift* means that we have

$$\begin{tikzcd}
\mathbf{Preord} \arrow[r, "F_\Gamma"] \arrow[d] & \mathbf{Preord} \arrow[d] \\
\mathbf{Set} \arrow[r, "F"'] & \mathbf{Set}
\end{tikzcd}$$

19.1 F_Γ-coalgebras

What is an F_Γ-coalgebra $(X, \leqslant, \zeta)$? It is an F-coalgebra (X, ζ) together with an endosimulation $(\leqslant)$ on (X, ζ).

We have a forgetful functor $U : \mathbf{Coalg}(F_\Gamma) \longrightarrow \mathbf{Coalg}(F)$, which maps $(X, \leqslant, \zeta)$ to (X, ζ).

U has a right adjoint $E : \mathbf{Coalg}(F) \longrightarrow \mathbf{Coalg}(F_\Gamma)$ which maps (X, ζ) to $(X, \text{similarity}, \zeta)$.

U has a left adjoint $\Delta : \mathbf{Coalg}(F) \longrightarrow \mathbf{Coalg}(F_\Gamma)$ which maps (X, ζ) to $(X, (=_X)\zeta)$

Since U and Δ are right adjoints, they preserve final objects.

Therefore a final F_Γ-coalgebra is an all-encompassing, extensional F-coalgebra, preordered by similarity.

We can use a final F_Γ-coalgebra to characterize both bisimilarity and similarity.

Let (X, ζ) and (Y, ξ) be F-coalgebras (transition systems). Let f and g be the anamorphisms from $(X, (=_X), \zeta)$ and $(Y, (=_Y), \xi)$.

Then for $x \in X$ and $y \in Y$

- x is bisimilar to y iff $f(x) = g(x)$
- x is similar to y iff $f(x) \leqslant g(x)$.

What if we take an all-encompassing system (e.g. a final F-coalgebra) and quotient by similarity?

Is this a final coalgebra?

See my FoSSaCS'11 paper!