Coinductive Predicates as Final Coalgebras

Milad Niqui* and Jan Rutten
Centrum Wiskunde & Informatica,
P. O. Box 94079, 1090 GB Amsterdam, The Netherlands
m.niqui@cwi.nl and jan.rutten@cwi.nl

Abstract

We show that coinductive predicates expressing behavioural properties of infinite objects can be themselves expressed as final coalgebras in a category of relations. The well-known case of bisimulation will simply be a special case of such final predicates. We will show how some useful pointwise and mixed properties of streams can be modelled in this way.

1 Introduction

Bisimulation is a widely used tool for proving the equivalent behaviour of infinite objects such as input/output systems and labelled transition systems. Although its original formulation was in the theory of processes and automata, later it was shown that maximal bisimulation — or *bisimilarity*— is tantamount to equality on the elements of final coalgebra. This leads to the *coinduction proof principle*: in order to prove that two elements of final coalgebra are equal, find a bisimulation between them. Furthermore, it was shown that for a certain class of functors, given any two coalgebras the set of all of bisimulation relations between them forms a complete lattice [14, Corollary 5.6]. It follows that bisimilarity is a post fixed point in the sense of Knaster-Tarski semantics.

Many have already observed that apart from equality there are other interesting binary relations on infinite objects which lead to other type of coinduction principles. Such principles are usually presented using Knaster-Tarski semantics. In this work we try to study this situation coalgebraically. Our starting point will be the well-known observation that bisimilarity itself is a final coalgebra for a different functor either in the same category (in the case of categorical models of dependent type theory [9, 8]) or in a different category (in the case of endofunctors on **Set** [10]). We will present variants of this observation: maximal bisimulation on different coalgebras i.e., not necessarily the final ones; and more generally arbitrary relations on elements of coalgebras.

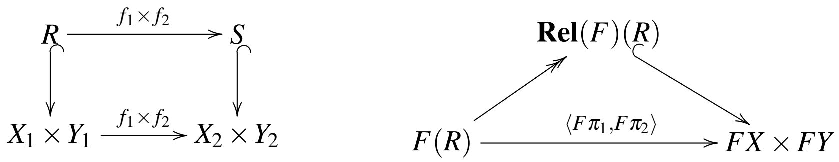
The motivation for this work is the use of coinductive predicates in theorem proving. Already in the short history of coinductive theorem proving it has become clear that most interesting behavioural properties usually need relatively complex coinductive predicates. Evidence can be found in the attempts to verify protocols [7], modalities [5] or even basic metric predicates on streams [3, 4]. This indicates that bisimilarity alone is not powerful enough for proving many properties of infinite objects. There are tools for the automatic generation of bisimulation [12]. Usually in such tools all other behavioural properties (e.g. the examples in Section 3) will be translated into equational problems so that they can be tackled using bisimulations. Depending on the problem domain, these translations might be costly or cumbersome; our aim is to make such tools applicable to automatic proofs for a larger class of properties without having to reduce each such property to an equational problem. To be more precise we would like to generalise the well-known hidden-algebraic fact 'Behavioural equivalence is bisimilarity' [13, 6] to a larger class of relations.

We restrict ourselves to a class of endofunctors on **Set**. However this work can be read in two different ways, in a categorical model of dependent type theory or in **Set** and relations on them.

*Supported by VENI grant nr. 639.021.714 from the Netherlands Organisation for Scientific Research (NWO).

2 Relation Lifting

A very general coalgebraic method for defining bisimulation is using *relation lifting* [11, Ch. 3]. Let **Rel** be the category of binary relations and relation-preserving maps, i.e., maps that make the leftmost diagram below commute (here $f_1 \times f_2$ on top is the obvious restriction of the bottom one).



Let $F\colon \mathbf{Set} \longrightarrow \mathbf{Set}$ be a functor. Then $\mathbf{Rel}(F)\colon \mathbf{Rel} \longrightarrow \mathbf{Rel}$, the *relation lifting of* F, is the functor taking a binary relation $\langle \pi_1, \pi_2\rangle\colon R \subseteq X \times Y$ to the image of $\langle F\pi_1, F\pi_2\rangle\colon FR \longrightarrow FX \times FY$ (see the rightmost diagram above). Given $f_1 \times f_2\colon R \longrightarrow S$ this **Rel**-functor is defined on morphisms as $\mathbf{Rel}(F)(f_1 \times f_2) := F(f_1) \times F(f_2)$. It is a well-known fact that relation lifting preserves equality, and thus it is geared towards proving equalities by constructing bisimulations. This make it unsuitable for working with arbitrary coinductive predicates. We consider a generalisation of relation lifting that can be used for a larger class of predicates.

Fix two sets X and Y. We denote by $\mathbf{Rel}_{XY}$ the lattice of subsets of $X \times Y$ considered as a subcategory of **Rel**, i.e., the objects are binary relations between X and Y and the morphisms are inclusion maps.

Let $\mathbf{rel}(F)\colon \mathbf{Rel}_{XY} \longrightarrow \mathbf{Rel}_{XY}$ be a *monotonic* functor such that for $R \subseteq X \times Y$ we have $\mathbf{rel}(F)(R) \subseteq F(X) \times F(Y)$. While the standard relation lifting takes R to a canonical subset $\mathbf{Rel}(F)(R) \subseteq F(X) \times F(Y)$, in our setting we deal with an arbitrary monotonic functor $\mathbf{rel}(F)$ taking relations on $X \times Y$ to relations on $F(X) \times F(Y)$. The reason is that the standard construction of bisimulations can, in a more generic way, be carried over to $\mathbf{rel}(F)$.

Note further that $\mathbf{Rel}(F)$ is an endofunctor on **Rel** while $\mathbf{rel}(F)$ is parametrised by X, Y and need not be defined globally. Obviously the theory we develop works for a $\mathbf{rel}(F)$ that is defined uniformly across **Rel**, so this is not a restriction. However, the local character of $\mathbf{rel}(F)$ allows the expression of finer properties. The examples in Section 3 demonstrate a globally defined $\mathbf{rel}(F)$ while in Section 4 we show examples where the local structure of $\mathbf{rel}(F)$ is needed.

For F-coalgebras $\alpha_X\colon X \longrightarrow FX$ and $\alpha_Y\colon Y \longrightarrow FY$, let $\tilde{F}_{XY}\colon \mathbf{Rel}_{XY} \longrightarrow \mathbf{Rel}_{XY}$ be the functor defined on objects as the inverse image of $\mathbf{rel}(F)$ alongside $\alpha_X \times \alpha_Y$ i.e.,

$$\tilde{F}_{\alpha_X\alpha_Y}(R) = \{\langle x,y\rangle \mid \langle \alpha_X(x), \alpha_Y(y)\rangle \in \mathbf{rel}(F)(R)\} \ ,$$

We usually drop the subscripts if they are understood from the context. As $\mathbf{rel}(F)$ is monotonic, so is $\tilde{F}$. In other words, $\tilde{F}$ is well-defined on the morphisms of $\mathbf{Rel}_{XY}$.

Note that in general we need not have $R \subseteq \tilde{F}(R)$ as this depends on the dynamics of α_X and α_Y. But if $R \subseteq \tilde{F}(R)$ then R with the inclusion map constitutes a $\tilde{F}$-coalgebra. Using Knaster-Tarski's fixed-point theorem, we can prove the following proposition.

Proposition 2.1. *The final coalgebra of* $\tilde{F}$ *exists in* $\mathbf{Rel}_{XY}$.

We denote the final $\tilde{F}$-coalgebras by $\nu\tilde{F}$. The final $\tilde{F}$-coalgebras correspond to coinductive predicates. This is because finality entails the equality

$$\tilde{F}(\nu\tilde{F}) = \nu\tilde{F} \ , \tag{1}$$

which means

$$\langle x,y\rangle \in \nu\tilde{F} \Leftrightarrow \langle \alpha_X(x), \alpha_Y(y)\rangle \in \mathbf{rel}(F)(\nu\tilde{F}) \ .$$

If one was to consider the above as the 'definition' of predicate $\nu\tilde{F}$ then the recursive occurrence of $\nu\tilde{F}$ as argument of $\mathbf{rel}(F)$ would supply the circularity evident in the coinductive predicates. Hence, by suitably choosing $\mathbf{rel}(F)$, α_X and α_Y, one can recover coinductive predicates as final $\tilde{F}$-coalgebras.

2.1 Bisimulation and Coinduction

Perhaps the most common example of a coinductive predicate is bisimulation between infinite objects. Any bisimulation between α_X and α_Y is a $\mathbf{Rel}(F)$-coalgebra $(R, \alpha_X \times \alpha_Y)$ in $\mathbf{Rel}_{\alpha_X \alpha_Y}$. Note that if we take $\mathbf{rel}(F) := \mathbf{Rel}(F)$ then $R \subseteq \tilde{F}(R)$ if and only if R is a bisimulation between α_X and α_Y [11]. This means that bisimulations are $\tilde{F}$-coalgebras with inclusion as transition map. Since $\tilde{F}$ is a monotonic functor it has a post fixed-point by Knaster-Tarski theorem, which is the maximal bisimulation [14]. The following proposition is then a reformulation of Proposition 2.1.

Proposition 2.2. *The final coalgebra of $\tilde{F}_{XY}$ exists in $\mathbf{Rel}_{\alpha_X \alpha_Y}$; its carrier is isomorphic with the maximal bisimulation between α_X and α_Y and its structure map is the identity inclusion map.*

Since equality includes any bisimulation relation on the carrier of an observable coalgebra (known also as a simple coalgebra) [14, Theorem 8.1] we have the following corollary.

Corollary 2.3 (Coinduction)**.**

i) Let $\langle \Omega^\circ, \alpha_{\Omega^\circ} \rangle$ be an observable F-coalgebra. If $\langle x, y \rangle \in \nu\tilde{F}_{\alpha_{\Omega^\circ}\alpha_{\Omega^\circ}}$ then $x = y$.

ii) Let $\langle \Omega, \alpha_\Omega \rangle = \nu F$ in **Set**. *If $\langle x, y \rangle \in \nu\tilde{F}_{\alpha_\Omega \alpha_\Omega}$ then $x = y$.*

Part (ii) of the corollary above is the basis for *type theoretic coinduction* that is used is systems such as *Coq* [1]. There, instead of the usual bisimulation-building technique, one shows that $\langle x, y \rangle$ is an element of the final coalgebra by constructing $x = y$ as a canonical element of the final coalgebra. This is possible because of the isomorphism in (1) which allows one to construct canonical elements using sufficiently guarded specifications.

3 Pointwise Coinductive Predicates on Streams

In this section we present some examples on streams to demonstrate that generalising the definition of $\mathbf{Rel}(F)$ to $\mathbf{rel}(F)$ indeed enables us to define more predicates.

In [11, § 3.1] an alternative inductive definition is given for relation lifting of polynomial functors which coincides with the aforesaid definition for $\mathbf{Rel}(F)$. Based on that inductive definition constant functors are lifted to the equality on their range, i.e., $\mathbf{Rel}(\Lambda X.A)(R) = \Delta_A$. Compared with the work in [11] this is what we modify: we replace equality by $\star \subseteq A \times A$, an arbitrary binary relation on A.

Let $F(X) := \mathbf{2} \times X$ and $\nu F = (\mathbf{2}^\omega, \langle \mathtt{hd}, \mathtt{tl} \rangle)$ be the set of binary streams as a final coalgebra. Now assume a relation $\star \subseteq \mathbf{2} \times \mathbf{2}$ and define for any sets X, Y and $R \subseteq X \times Y$:

$$\mathbf{Rel}_\star(F)(R) = \{\langle \langle b_1, x \rangle, \langle b_2, y \rangle \rangle \mid \langle b_1, b_2 \rangle \in \star \wedge xRy\} \ .$$

Since $\mathbf{Rel}_\star$ is monotonic, by taking $\mathbf{rel}(F) := \mathbf{Rel}_\star(F)$ we can define the corresponding $\tilde{F}$ from Section 2 and apply Proposition 2.1 to get its final coalgebra. Clearly the description of this final coalgebra depends on the relation $\star$. For the case where the underlying coalgebras α_X, α_Y are the final coalgebra of streams

[1] In *Coq* and other intensional type theories this does not entail $x = y$ *inside* the system, but this issue is beyond the present paper.

this has a simple solution. The answer is given in the proposition below, for which we need some definitions. Define the binary relation $\circledast$ on streams as

$$\circledast := \{\langle \sigma, \tau \rangle \mid \forall n, \langle \mathtt{hd}(\mathtt{tl}^n(\sigma)), \mathtt{hd}(\mathtt{tl}^n(\tau)) \rangle \in \star\} \ ,$$

i.e., two streams σ, τ belong to $\circledast$ if and only if they are pointwise in relation $\star$. Note that

$$\begin{aligned} \tilde{F} \circledast &= \{\langle \sigma, \tau \rangle \mid \langle \langle \mathtt{hd}, \mathtt{tl} \rangle(\sigma), \langle \mathtt{hd}, \mathtt{tl} \rangle(\tau) \rangle \in \mathbf{Rel}_\star(F)(\circledast)\} \\ &= \{\langle \sigma, \tau \rangle \mid \langle \mathtt{hd}(\sigma), \mathtt{hd}(\tau) \rangle \in \star \wedge \langle \mathtt{tl}(\sigma), \mathtt{tl}(\tau) \rangle \in \circledast\} \end{aligned}$$

Note that $\circledast \subseteq \tilde{F} \circledast$; hence we can define $\alpha_\circledast : \ \circledast \longrightarrow \tilde{F} \circledast$ as the inclusion map. We have the following proposition.

Proposition 3.1. $\nu \tilde{F} = (\circledast, \alpha_\circledast)$.

The proof is a straightforward induction and resembles the proof of finality of the set of streams in [1].

If $R \subseteq \tilde{F}(R)$ we shall call R a $\star$-*simulation*. Clearly $\circledast$ is a $\star$-simulation on $\mathbf{2}^\omega$. Note that if R is $\star$-simulation and $\langle \sigma, \tau \rangle$ in R, then $\langle \mathtt{tl}^n(\sigma), \mathtt{tl}^n(\tau) \rangle \in R$ for all n. From this fact we can obtain the following principle.

Proposition 3.2 ($\star$-Coinduction)**.** *Relation $\circledast$ is the maximal $\star$-simulation relation on $\mathbf{2}^\omega$. I.e., in order to prove two streams are in $\circledast$ it suffices if we find a $\star$-simulation between them.*

Example 3.3.

i) Taking $\star := \Delta_\mathbf{2}$ we can recover relation lifting of [11], as well as bisimilarity and the coinduction proof principle.

ii) Taking $\star$ to be $\neq$ i.e., $\star := \{\langle 0, 1 \rangle, \langle 1, 0 \rangle\}$ we get the pointwise *inequality* between streams as a coinductive predicate. I.e., the $\circledast := \not\asymp$ where $\sigma \not\asymp \tau$ if and only if $\mathtt{hd}(\mathtt{tl}^n(\sigma)) \neq \mathtt{hd}(\mathtt{tl}^n(\sigma))$ for all n. Assume constant streams $\mathtt{zeros}$, $\mathtt{ones}$ to be defined as

$$\begin{aligned} &\mathtt{hd}(\mathtt{zeros}) := 0 \ , \quad \mathtt{tl}(\mathtt{zeros}) := \mathtt{zeros} \ ; \\ &\mathtt{hd}(\mathtt{ones}) := 1 \ , \quad \mathtt{tl}(\mathtt{ones}) := \mathtt{ones} \ . \end{aligned}$$

Then since $R_{\neq} := \{\langle 0 :: \mathtt{zeros}, 1 :: \mathtt{ones} \rangle\}$ is a $\neq$-simulation by Proposition 3.2 we have $\mathtt{zeros} \not\asymp \mathtt{ones}$.

iii) Similar to above, by taking $\star := \{\langle 0, 1 \rangle\}$ (resp. $\star := \{\langle 0, 0 \rangle, \langle 0, 1 \rangle, \langle 1, 1 \rangle\}$) we get $\prec$ the pointwise *less than* (resp. $\preccurlyeq$ *less or equal*) relation between streams as a coinductive predicate. Again one can show by Proposition 3.2

$$\mathtt{zeros} \prec \mathtt{ones} \ , \qquad \mathtt{zeros} \preccurlyeq \mathtt{ones} \ , \qquad \mathtt{zeros} \preccurlyeq \mathtt{zeros} \ .$$

Note that the relations in the example above are *not* the same as *simulations* in the sense of [10]. Simulation (and *lax* relation lifting), albeit itself a greatest fixed point [10, Lemma 5.1], is based on an order on the functor while in our pointwise comparison we use an arbitrary binary relation on data which is not necessarily an order relation.

Proposition 3.2 is useful in that it mimics the ordinary coinduction proof principle. However, one can also directly use the finality of $\circledast$ to construct its elements in their canonical form. This leads to another instance of type theoretic coinduction.

We conclude this section by pointing out that the above lifting of pointwise relations to the stream level can be done for relations with different arity. In fact it is straightforward to obtain the counterpart of Proposition 3.2 for n-ary relations. For example one can consider a ternary relation $\star_3 := \{\langle 0,0,0\rangle, \langle 0,1,1\rangle, \langle 1,0,1\rangle, \langle 1,1,1\rangle\}$ that leads to the relation $\curlyvee$ corresponding to the pointwise disjunction of streams. This is specially useful in proving properties in stream calculus [15], because there we deal with causal functions and examining the components pointwise will usually suffice.

4 Mixed Coinductive Predicates on Streams

The pointwise coinductive predicates, though useful in many cases, have a rather restricted shape that limits their expressiveness. While our Proposition 2.1 is quite general, it is not always easy to find a simple description of the elements of the final coalgebra as in Proposition 3.1. In this section, still working with binary streams, we show some more intricate coinductive predicates that are useful in practise. In particular we show two examples from [2] which are used in a coinductive timed stream semantics of channel-based coordination[2].

Fix nonempty relations $\star_1, \star_2 \subseteq \mathbf{2}\times\mathbf{2}$. Assume (X,α_X) and (Y,α_Y) are F-coalgebras. Let

$$\begin{aligned}\mathbf{Rel}_{\star_1\star_2}(F)(R) = &\{\langle\langle b_1,x\rangle,\langle b_2,y\rangle\rangle \mid \langle b_1,b_2\rangle\in\star_1 \wedge \exists t\in\alpha_Y^{-1}\langle b_2,y\rangle,\ \langle x,t\rangle\in R\}\ \cup\\ &\{\langle\langle b_1,x\rangle,\langle b_2,y\rangle\rangle \mid \langle b_1,b_2\rangle\in\star_2 \wedge \exists t\in\alpha_X^{-1}\langle b_1,x\rangle,\ \langle t,y\rangle\in R\}\ .\end{aligned}$$

Then $\mathbf{Rel}_{\star_1\star_2}(F)$ is a monotonic endofunctor on $\mathbf{Rel}_{XY}$. Again the corresponding $\tilde{F}$ and its final coalgebra can be formed according to Section 2, but this general form is too complex to be useful. The special case where $\alpha_X = \alpha_Y = \langle\mathtt{hd},\mathtt{tl}\rangle$ leads to some simplification. In that case,

$$\begin{aligned}\tilde{F}(R) = &\{\langle\sigma,\tau\rangle \mid \langle\mathtt{hd}(\sigma),\mathtt{hd}(\tau)\rangle\in\star_1\wedge\langle\mathtt{tl}(\sigma),\tau\rangle\in R\}\ \cup\\ &\{\langle\sigma,\tau\rangle \mid \langle\mathtt{hd}(\sigma),\mathtt{hd}(\tau)\rangle\in\star_2\wedge\langle\sigma,\mathtt{tl}(\tau)\rangle\in R\}\ .\end{aligned}$$

Instantiating with $\star_1 = \{\langle 0,1\rangle\}$, $\star_2 = \{\langle 1,0\rangle\}$ one can observe that the final coalgebra $\nu\tilde{F}$ consists of the set of binary streams that satisfy the $\bowtie$ relation as defined in [2]. In short, if $\sigma\bowtie\tau$ and if both σ and τ are interpreted as time streams corresponding to events on the two ports of a channel, then σ and τ are completely asynchronous. Again we call relation R a $\bowtie$-simulation if $R\subseteq\tilde{F}(R)$ and we will have a counterpart of Proposition 3.2. I.e., in order to prove that two time streams are asynchronous it suffices to find a $\bowtie$-simulation between them.

Next example concerns the *merge* connective in [2], which captures the behaviour of a merger channel with two inputs and one output, and merges its two input streams to form the output. Here we work with ternary relations.

$$\begin{aligned}\mathbf{Rel}_{merge}(F)(R) = &\{\langle\langle b_1,x\rangle,\langle b_2,y\rangle,\langle b_3,z\rangle\rangle \mid \langle b_1,b_2\rangle\in\star_1\wedge\langle b_1,b_3\rangle\in\star_2\wedge\exists t{\in}\alpha_Y^{-1}\langle b_2,y\rangle,\ \langle x,t,z\rangle\in R\}\ \cup\\ &\{\langle\langle b_1,x\rangle,\langle b_2,y\rangle,\langle b_3,z\rangle\rangle \mid \langle b_1,b_2\rangle\in\star_3\wedge\langle b_2,b_3\rangle\in\star_4\wedge\exists t{\in}\alpha_X^{-1}\langle b_1,x\rangle,\ \langle t,y,z\rangle\in R\}\ .\end{aligned}$$

Assuming $\alpha_\_$'s are all the structure map of the final coalgebra of streams we have

$$\begin{aligned}\tilde{F}(R) = &\{\langle\sigma_1,\sigma_2,\tau\rangle \mid \langle\mathtt{hd}(\sigma_1),\mathtt{hd}(\sigma_2)\rangle\in\star_1\wedge\langle\mathtt{hd}(\sigma_1),\mathtt{hd}(\tau)\rangle\in\star_2\wedge\langle\mathtt{tl}(\sigma_1),\sigma_2,\mathtt{tl}(\tau)\rangle\in R\}\ \cup\\ &\{\langle\sigma_1,\sigma_2,\tau\rangle \mid \langle\mathtt{hd}(\sigma_1),\mathtt{hd}(\sigma_2)\rangle\in\star_3\wedge\langle\mathtt{hd}(\sigma_2),\mathtt{hd}(\tau)\rangle\in\star_4\wedge\langle\sigma_1,\mathtt{tl}(\sigma_2),\mathtt{tl}(\tau)\rangle\in R\}\ .\end{aligned}$$

[2] Timed data streams in [2] have a data component as well. For brevity, here we do not tackle the data and only deal with the time. Another simplification is that we use binary time but one could repeat this for $F(X) = \mathbb{R}^+\times X$.

Instantiating with $\star_1 = \{\langle 0,1\rangle\}$, $\star_3 = \{\langle 1,0\rangle\}$, $\star_2 = \star_4 = \mathbf{\Delta_2}$ we obtain a coinductive predicate describing the behaviour of the merger channel. Furthermore we obtain a notion of *merge*-simulation and a corresponding coinduction principle. This enables us to prove, by finding a *merge*-simulation, that three time streams correspond to events on the ports of a merger.

Our last example illustrates a predicate $\asymp$ on binary streams such that (denoting $\mathtt{hd}(\mathtt{tl}^n(\sigma))$ by σ_n):

$$\sigma \asymp \tau := \forall n, \sigma_{2n} \leq \tau_{2n+1} \leq \sigma_{2n+2} \ \wedge \ \tau_{2n} \leq \sigma_{2n+1} \leq \tau_{2n+2} \ .$$

Hence $\sigma \asymp \tau$ means that $\sigma_0 :: \tau_1 :: \sigma_2 :: \tau_3 \cdots$ and $\tau_0 :: \sigma_1 :: \tau_2 :: \sigma_3 \cdots$ are both non-decreasing streams. Compared to the previous examples the predicate $\asymp$ examines larger initial segments of the streams and hence our candidate for $\mathbf{rel}(F)$ should observe deeper iterations of coalgebra. Let

$$\begin{aligned}\mathbf{Rel}_{\asymp}(F)(R) = \{\langle\langle b_1,x\rangle,\langle b_2,y\rangle\rangle \mid\ & \alpha_X(x) = \langle b_1',x'\rangle \ \wedge \alpha_X(x') = \langle b_1'',x''\rangle \ \wedge \\ & \alpha_Y(y) = \langle b_2',y'\rangle \ \wedge \alpha_Y(y') = \langle b_2'',y''\rangle \implies \\ & b_1 \leq b_2' \leq b_1'' \ \wedge \ b_2 \leq b_1' \leq b_2'' \ \wedge \ \langle x'',y''\rangle \in R\} \ .\end{aligned}$$

Then $\mathbf{Rel}_{\asymp}(F)$ is monotonic and we can form $\tilde{F}$ and its final coalgebra. Assuming $\alpha_X = \alpha_Y = \langle \mathtt{hd}, \mathtt{tl}\rangle$ we obtain

$$\tilde{F}(R) = \{\langle\sigma,\tau\rangle \mid \sigma_0 \leq \tau_1 \leq \sigma_2 \ \wedge \ \tau_0 \leq \sigma_1 \leq \tau_2 \ \wedge \ \langle \mathtt{tl}^3(\sigma), \mathtt{tl}^3(\tau)\rangle \in R\} \ .$$

Then $\langle\sigma,\tau\rangle \in \nu\tilde{F} \Leftrightarrow \sigma \asymp \tau$ and in order to prove that $\sigma \asymp \tau$ we should find a relation R such that $R \subseteq \tilde{F}(R)$ and that $\langle\sigma,\tau\rangle \in R$. As an example let the streams zos, ozs be defined as

$$\begin{aligned}&\mathtt{hd}(\mathtt{zos}) := 0 \ , \quad \mathtt{hd}(\mathtt{tl}(\mathtt{zos})) := 1 \ , \quad \mathtt{tl}^2(\mathtt{zos}) := \mathtt{zos} \ ; \\ &\mathtt{hd}(\mathtt{ozs}) := 1 \ , \quad \mathtt{hd}(\mathtt{tl}(\mathtt{ozs})) := 0 \ , \quad \mathtt{tl}^2(\mathtt{ozs}) := \mathtt{ozs} \ .\end{aligned}$$

Then taking $R_{\asymp} := \{\langle \mathtt{zos},\mathtt{ozs}\rangle, \langle \mathtt{ozs},\mathtt{zos}\rangle\}$, and considering that by ordinary coinduction

$$\mathtt{zos} = 0 :: \mathtt{ozs} \ , \quad \mathtt{ozs} = 1 :: \mathtt{zos} \ ,$$

we obtain $\tilde{F}(R_{\asymp}) = R_{\asymp}$. Hence $\mathtt{zos} \asymp \mathtt{ozs}$.

5 Conclusion & Further Work

The fact that we can describe such predicates as final coalgebras in **Rel** has more usage. By having a final model in hand coinductive proofs will essentially turn into finding functions between various final coalgebras. Hence we can use several type of coinductive definition schemes (e.g. coiteration, corecursion and their generalisations) for more complicated proofs. For example using the examples developed in Sections 3–4 one can prove

$$\begin{aligned}&\forall \sigma\tau,\ \sigma \prec \tau \implies \sigma \not\asymp \tau \\ &\forall \sigma_1\sigma_2\tau,\ \mathit{merge}(\sigma_1,\sigma_2,\tau) \implies \sigma_1 \bowtie \sigma_2 \ .\end{aligned}$$

The first implication is in fact a function between final coalgebras $\circledast_< \to \circledast_{\neq}$ that can be defined using the ordinary coiteration scheme.

In the future, we plan to work on automating the generation of various types of $\star$-simulation relations in the tools that are used for automatic generation of bisimulations [12]. This requires a thorough reformulation of hidden-algebraic machinery of behavioural equivalence in a more general way. Recall that two streams are behaviourally equivalent if and only if $\mathtt{hd}(\mathtt{tl}^n(\sigma)) = \mathtt{hd}(\mathtt{tl}^n(\sigma))$ for all n; and that this implies bisimilarity. Looking back at the definition of $\circledast$ we observe that it captures a notion of 'behaviourally being in relation $\star$' which then will imply $\star$-similarity. Our aim is to make this more precise by working in the categorical models of hidden-algebra [6].

References

[1] P. Aczel. Algebras and coalgebras. In R. C. Backhouse, R. L. Crole, and J. Gibbons, eds., *Revised Lectures from Int. Summer School and Wksh. on Algebraic and Coalgebraic Methods in the Mathematics of Program Construction (Oxford, Apr. 2000)*, v. 2297 of *Lect. Notes in Comput. Sci.*, pp. 79–88. Springer, 2002.

[2] F. Arbab and J. J. M. M. Rutten. A coinductive calculus of component connectors. In M. Wirsing, D. Pattinson, and R. Hennicker, eds., *Revised Selected Papers from 16th Int. Wksh. on Algebraic Development Techniques, WADT 2002 (Frauenchiemsee, Sept. 2002)*, v. 2755 of *Lect. Notes in Comput. Sci.*, pp. 34–55. Springer, 2003.

[3] Y. Bertot. Filters on coinductive streams, an application to Eratosthenes' sieve. In P. Urzyczyn, ed., *Proc. of 7th Int. Conf. on Typed Lambda Calculi and Applications, TLCA 2005 (Nara, Apr. 2005)*, v. 3461 of *Lect. Notes in Comput. Sci.*, pp. 102–115. Springer, 2005.

[4] Y. Bertot. Affine functions and series with co-inductive real numbers. *Math. Struct. in Comput. Sci.*, 17(1):37–63, 2007.

[5] V. Capretta. Common knowledge as a coinductive modality. In E. Barendsen, V. Capretta, H. Geuvers, and M. Niqui, eds., *Reflections on Type Theory, λ-Calculus, and the Mind: Essays Dedicated to Henk Barendregt on the Occasion of his 60th Birthday*, pp. 51–61. Radboud University Nijmegen, 2007.

[6] C. Cîrstea. Coalgebra semantics for hidden algebra: Parameterised objects an inheritance. In F. Parisi-Presicce, ed., *Selected Papers from 12th Int. Wksh. on Algebraic Development Techniques, WADT '97 (Tarquinia, June 1997)*, v. 1376 of *Lect. Notes in Comput. Sci.*, pp. 174–189. Springer, 1997.

[7] E. Giménez. *Un Calcul de Constructions Infinies et son Application a la Verification des Systemes Communicants*. PhD thesis, Ecole Normale Supérieure de Lyon, 1996.

[8] P. Hancock and A. Setzer. Interactive programs and weakly final coalgebras in dependent type theory. In L. Crosilla and P. Schuster, eds., *From Sets and Types to Topology and Analysis: Towards Practicable Foundations for Constructive Mathematics*, v. 48 of *Oxford Logic Guides*, pp. 115–134. Oxford Univ. Press, 2005.

[9] C. Hermida and B. Jacobs. Structural induction and coinduction in a fibrational setting. *Inform. and Comput.*, 145(2):107–152, 1998.

[10] J. Hughes and B. Jacobs. Simulations in coalgebra. *Theor. Comput. Sci.*, 327(1–2):71–108, 2004.

[11] B. Jacobs. *Introduction to Coalgebra: Towards Mathematics of States and Observations*. Book draft, 2005. Available at `http://www.cs.ru.nl/B.Jacobs/CLG/JacobsCoalgebraIntro.pdf`,

[12] D. Lucanu and G. Roşu. CIRC: a circular coinductive prover. In T. Mossakowski, U. Montanari, and M. Haveraaen, eds., *Proc. of 2nd Int. Conf. on Algebra and Coalgebra in Computer Science, CALCO 2007 (Bergen, Aug. 2007)*, v. 4624 of *Lect. Notes in Comput. Sci.*, pp. 372–378. Springer, 2007.

[13] G. Malcolm. Behavioural equivalence, bisimulation, and minimal realisation. In M. Haveraaen, O. Owe, and O.-J. Dahl, eds., *Selected Papers from 11th Wksh. on Specification Abstract Data Types, Joint with 8th Compass Wksh., ADT/COMPASS 1995 (Oslo, Sept. 1995)*, v. 1130 of *Lect. Notes in Comput. Sci.*, pp. 359–378. Springer, 1996.

[14] J. J. M. M. Rutten. Universal coalgebra: a theory of systems. *Theor. Comput. Sci.*, 249(1):3–80, 2000.

[15] J. J. M. M. Rutten. A coinductive calculus of streams. *Math. Struct. in Comput. Sci.*, 15(1):93–147, 2005.