From Set-theoretic Coinduction to Coalgebraic Coinduction: some results, some problems

Marina Lenisa *

Laboratory for the Foundations of Computer Science
University of Edinburgh, Scotland.
`lenisa@dcs.ed.ac.uk`

December 15, 1999

Abstract

We investigate the relation between the *set-theoretical* description of coinduction based on Tarski Fixpoint Theorem, and the *categorical* description of coinduction based on *coalgebras*. In particular, we examine set-theoretic generalizations of the *coinduction proof principle*, in the spirit of Milner's *bisimulation "up-to"*, and we discuss categorical counterparts for these. Moreover, we investigate the connection between these and the equivalences induced by T*-coiterative functions*. These are morphisms into *final coalgebras*, satisfying the T*-coiteration scheme*, which is a generalization of both the *coiteration* and the *corecursion scheme*. We generalize Rutten's transformation from coalgebraic bisimulations to set-theoretic bisimulations, in order to cover also the case of bisimulations "up-to". A list of examples of set-theoretic coinductive specifications which appear not to be easily expressible in coalgebraic terms are discussed.

Introduction

Coinductive definitions and *coinduction proof principles* are a natural tool for defining and reasoning on *infinite* and *circular* objects, such as streams, exact reals, processes. See e.g. [Mil83, Coq94, HL95, BM96, Fio96, Len96, Pit96, Rut96, HJ98, HLMP98, Len98] for various approaches to infinite objects based on coinduction. Many of such objects and concepts arise in connection with a *maximal fixed point* construction of some kind. One of the advantages offered by the coinductive approach with respect to others based on domain theory or metric semantics, is that it allows for a *simple, operationally-based, implementation-independent* treatment of infinite objects, without requiring any heavy mathematical overhead. A purely set-theoretical approach, however, often appears quite ad-hoc, just think, for example, of how one would prove set-theoretically the existence of a *coiterative* function into streams.

In recent years, a *categorical* explanation of coinduction has appeared, based on the notion of *coalgebra*, see e.g. ([Acz88, AM89, Acz93, RT93, RT94, Tur96, TP97, Len98]);

*Work supported by Esprit Working Group "Types", MURST'97 Cofin. "Sistemi Formali..." grant, TMR Linear FMRX-CT98-0170.

we shall refer to it as *coalgebraic coinduction*. Coalgebraic coinduction has proved to be extremely fertile ([HL95, Jac96, Len96, Rut96, Jac97, RV97, HLMP98, Mos, Len99]). For instance, it has spurred the development of *Final Semantics*, a methodology for understanding the correspondence between syntax and operational semantics of programming languages. Whenever coalgebraic coinduction is successful, it overcomes many of the defects of set-theoretic coinduction. It explains coinductive proof techniques uniformly and suggestively. It allows to treat simultaneously definitions by corecursion and to phrase proofs by coinduction in a more principled and uniform way.

We feel, however, that there is still a wide range of contexts where set-theoretic coinductive tools have not yet been explained coalgebraically. Moreover, very few attempts have been made to formulate precisely the correspondence between set-theoretic and coalgebraic coinduction, and the scope of the latter.

In this paper, which expands ideas in [Len98], we offer some contributions along these directions of research, so far little explored.

First, we introduce various generalizations of the classical set-theoretical coinduction principle based on Tarski Fixpoint Theorem, which go in the direction of Milner's bisimulation "up-to" principle ([Mil83]). We call these proof principles *coinduction principles "up-to"*. The practical interest of coinduction principles "up-to" lies in the fact that they allow to factorize the complexity of proofs by coinduction. In particular, they allow to reduce the size of the relations to be exhibited in a proof by coinduction. An important example of coinduction "up-to" is *coinduction "up-to-context"*, which we shall discuss in the paper.

Then we try to develop coalgebraic counterparts of these set-theoretic coinduction principles "up-to". In particular, we present a coalgebraic version of the set-theoretic principle "up-to-T", for T suitable operator on relations. We call this categorical version Coalgebraic Coinduction "up-to-T", for T suitable monad. This is based on the new notion of F-bisimulation "up-to-T". The Coalgebraic Coinduction "up-to-T" is put to use in order to get a *proof principle* for reasoning on equivalences induced by *T-coiterative morphisms*. I.e. morphisms into final coalgebras defined by the *T-coiteration scheme*. This generalizes both the *coiteration* and the *corecursion scheme* (see [Geu92]). The latter is dual to the (primitive) recursion scheme. The T-coiteration scheme allows to capture many interesting functions into final coalgebras, which escape the pure coiteration scheme (see Section 3.1.1 for an example). Moreover, there are cases in which the definition by generalized coiteration is more natural than that by simple coiteration. In particular, we define a *weak semantics* for CCS processes, i.e. a semantics inducing the *weak congruence*, by T-coiteration, for a suitable monad T. This semantics is alternative to that defined by coiteration in [Acz93]. Generalized coiteration schemes illustrate the advantages that a coalgebraic description offers w.r.t. a set-theoretical one, as far as uniformity and generality.

The relations between Tarski's coinduction principle and the categorical principle based on F-bisimulations can be formalized precisely. Namely, for all functors which preserve weak pullbacks, one can derive set-theoretic coinduction principles from categorical coinduction principles ([Rut98]). We show that the translation from coalgebraic to set-theoretic coinduction is *compositional*. Moreover, we generalize Rutten's translation to coinduction "up-to". Going the other way round, i.e. deriving categorical principles from set-theoretic ones, seems to be very problematic. The main reason for this is that coalgebraic coinduction essentially captures equivalences induced by coiterative morphisms, while coalgebraic descriptions of other predicates on coinductive types, even of other equivalences, appear problemaic. In this paper, we provide some critical situations which seem to indicate limitations of the coalgebraic approach.

The paper is organized as follows. In Section 1, we recall the classical coinduction principle deriving from Tarski's characterization of greatest fixed points of monotone operators, and we introduce a number of (possibly new) set-theoretic coinduction principles "up-to". In Section 2, we present the categorical framework based on coalgebras for describing coinduction and coiteration. In Section 3, we introduce the *T-coiteration scheme*, and we present a categorical counterpart for the set-theoretic coinduction principle "up-to-T" introduced in Section 1. Moreover, we derive a *sound* proof principle for establishing equivalences induced by T-coiterative morphisms. In Section 4, we study the relations between the set-theoretic coinduction principles of Section 1, and the categorical coinduction principles based on F-bisimulations of Sections 2 and 3.

The author is grateful to Furio Honsell, Jan Rutten, Davide Sangiorgi, and Daniele Turi for useful discussions on some of the issues in this paper, and is especially thankful to John Power for illuminating remarks.

1 Set-theoretic Coinduction

In this section, we list a number of set-theoretic coinduction principles, ranging from the basic coinduction principle, originally used by Milner in [Mil83] for reasoning on CCS processes, to the principles "up-to" ([San98, Len98]), which generalize the idea behind the notion of Milner's *bisimulation "up-to"*.

For simplicitly, we discuss monotone operators on binary relations, but all the results apply more in general to monotone operators on complete lattices.

Throughout this section, $\Phi : \mathcal{P}(X \times Y) \to \mathcal{P}(X \times Y)$ will denote a *monotone* operator over the complete lattice of set-theoretic binary relations on the cartesian product of two sets X and Y, and $\approx_\Phi$ will denote the greatest fixed point of Φ.

Theorem 1 (Coinduction Principle (Tarski)) *The following principle is* sound*:*

$$\frac{x\ \mathcal{R}\ y \quad \mathcal{R} \subseteq \Phi(\mathcal{R})}{x \approx_\Phi y} \ .$$

The Coinduction Principle is also complete in the sense that

$$x \approx_\Phi y \implies \exists\, \mathcal{R}\, .\ (x\ \mathcal{R}\ y \land\ \mathcal{R} \subseteq \Phi(\mathcal{R}))\ .$$

As usual, we call *Φ-bisimulation* a relation $\mathcal{R}$ which satisfies the second premise of the principle of Coinduction.

Coinduction principles are the more useful, the easier it is to show the inclusion in the premise, and, consequently, the smaller the complexity of the relation $\mathcal{R}$ to exhibit. It is therefore natural to look for alternative characterizations of maximal fixed points, possibly exploiting special properties of the operator Φ, which allow to relax the condition $\mathcal{R} \subseteq \Phi(\mathcal{R})$.

1.1 Coinduction "up-to-T"

A simple and natural example of coinduction "up-to" is given by the following theorem:

Theorem 2 (Coinduction "up-to-$\cup$") *Let $\overline{\mathcal{R}} \in \mathcal{P}(X \times Y)$ be such that $\overline{\mathcal{R}} \subseteq \approx_\Phi$. Then*
i)

$$\approx_\Phi = \bigcup \{\mathcal{R} \mid \mathcal{R} \subseteq \Phi(\mathcal{R} \cup \overline{\mathcal{R}})\} \ .$$

ii) The following principle is sound and complete:

$$\frac{x\ \mathcal{R}\ y \quad \mathcal{R} \subseteq \Phi(\mathcal{R} \cup \overline{\mathcal{R}})}{x \approx_\Phi y} \ .$$

Interesting instances of the above principle arise when $\overline{\mathcal{R}}$ is taken to be $\approx_\Phi$, or the identity relation, if $\approx_\Phi$ is reflexive.

Now we discuss a more substantial generalization of the coinduction principle, in the spirit of Milner's bisimulation "up-to", which we call principle of *Coinduction "up-to-T"*, and which subsumes the principle of Coinduction "up-to-$\cup$". We give two versions of the principle of Coinduction "up-to-T". The first is an immediate modification of Sangiorgi's principle ([San98]) for labelled transition systems, and it is a *sound* (but not in general *complete*) principle. The latter is obtained from the previous one by strengthening the conditions on the operator T, in order to get also completeness.

Theorem 3 (Coinduction "up-to-T", I, [San98]) *Let $T : \mathcal{P}(X \times Y) \to \mathcal{P}(X \times Y)$. If T is* respectful, *i.e., for all $\mathcal{R}, \mathcal{R}'$,*

$$(\mathcal{R} \subseteq \mathcal{R}' \ \wedge \ \mathcal{R} \subseteq \Phi(\mathcal{R}')) \implies (T(\mathcal{R}) \subseteq T(\mathcal{R}') \ \wedge \ T(\mathcal{R}) \subseteq \Phi \circ T(\mathcal{R}')) \ ,$$

then the following principle is sound:

$$\frac{x\ \mathcal{R}\ y \quad \mathcal{R} \subseteq (\Phi \circ T)(\mathcal{R})}{x \approx_\Phi y} \ .$$

Theorem 4 (Coinduction "up-to-T", II) *Let $T : \mathcal{P}(X \times Y) \to \mathcal{P}(X \times Y)$. If T satisfies the following properties*
1) T is a monotone operator on the complete lattice $(\mathcal{P}(X \times Y), \subseteq)$,
2) for all $\mathcal{R} \in \mathcal{P}(X \times Y)$, $\mathcal{R} \subseteq T(\mathcal{R})$,
3) for all $\mathcal{R} \in \mathcal{P}(X \times Y)$, $(T \circ \Phi)(\mathcal{R}) \subseteq (\Phi \circ T)(\mathcal{R})$,
then
i)

$$\approx_\Phi = \bigcup \{\mathcal{R} \mid \mathcal{R} \subseteq (\Phi \circ T)(\mathcal{R})\} \ .$$

ii) The following principle is sound and complete:

$$\frac{x\ \mathcal{R}\ y \quad \mathcal{R} \subseteq (\Phi \circ T)(\mathcal{R})}{x \approx_\Phi y} \ .$$

Proof. i) Let $\mathcal{R} \subseteq \Phi(\mathcal{R})$. Then, by hypothesis 2 and by monotonicity of Φ, $\mathcal{R} \subseteq \Phi(\mathcal{R}) \subseteq \Phi(T(\mathcal{R}))$.
Vice versa, if $\mathcal{R} \subseteq \Phi \circ T(\mathcal{R})$, we have to show that $\exists\, \mathcal{S}$ such that $\mathcal{R} \subseteq \mathcal{S} \ \wedge \ \mathcal{S} \subseteq \Phi(\mathcal{S})$. Consider the following inductively defined sequence $\{\mathcal{R}_n\}_{n \geq 0}$:

$$\mathcal{R}_0 = \mathcal{R}$$
$$\mathcal{R}_{n+1} = T(\mathcal{R}_n) \ .$$

We prove by induction on n that $\mathcal{R}_n \subseteq \Phi(\mathcal{R}_{n+1})$:
For $n = 0$ the thesis is immediate, since $\mathcal{R} \subseteq \Phi \circ T(\mathcal{R})$ by hypothesis.
Let $n > 0$:
$\mathcal{R}_n = T(\mathcal{R}_{n-1})$
$\subseteq T \circ \Phi(\mathcal{R}_n)$, by induction hypothesis and monotonicity of T,
$= \Phi \circ T(\mathcal{R}_n)$, by hypothesis 3,
$= \Phi(\mathcal{R}_{n+1})$, by definition of the sequence.
Hence, taking $\mathcal{S} = \bigcup_n \mathcal{R}_n$, we have $\mathcal{R} \subseteq \mathcal{S}$ and $\mathcal{S} \subseteq \Phi(\mathcal{S})$.
ii) Immediate consequence of item i) of this theorem. $\square$

If we take both X, Y, in Theorem 3 or in Theorem 4, to be the domain of CCS processes, Φ to be the operator corresponding to strong bisimulation, and we take T to be defined by $T(\mathcal{R}) = \approx_\Phi \circ \mathcal{R} \circ \approx_\Phi$, we have that a relation $\mathcal{R}$ such that $\mathcal{R} \subseteq (\Phi \circ T)(\mathcal{R})$ is a bisimulation "up-to" in the sense of Milner.

Sangiorgi's principle is complete when considered over all respectful operators, i.e. the union of all bisimulations "up-to-T" is $\approx_\Phi$, if T ranges over all respectful opeartors. But, for each fixed T, it is only sound. A simple counterexample is the following. If the operator T is the constant operator equal to the least fixed point of Φ, $\sim_\Phi$, and moreover $\sim_\Phi \neq \approx_\Phi$, then $\approx_\Phi \neq \bigcup\{\mathcal{R} \mid \mathcal{R} \subseteq (\Phi \circ T)(\mathcal{R})\}$. Notice that the respectfulness condition is already implied by the sole hypotheses 1 and 3 of Theorem 4. If we drop hypothesis 2 in Theorem 4, then again we can prove only soundness, but not completeness of the Coinduction Principle "up-to-T". The weaker condition of Sangiorgi's principle is devised especially to capture important operators T, such as the *contextual closure* operator, which does not satisfy condition 3 of Theorem 4. This is a powerful tool for reasoning on operational equivalences of programming languages, ranging from CCS-like processes algebras, to π-calculus, to λ-calculus. We formally introduce this operator for CCS processes in the following section.

The contextual closure operator. We focus on a simple algebra of CCS-like processes. We briefly review the syntax and the operational semantics of CCS without restriction and relabelling, for simplicitly (but they could be considered without adding any substantial complication).

Definition 1 (CCS Processes) *Let $\mathcal{A}$ be an infinite set of* atomic actions *such that the special* silent action $\tau \in A$, *and for each* $a \in \mathcal{A}$, *also* $\overline{a} \in \mathcal{A}$, *and* $\overline{\overline{a}} = a$. *Let* PVar *be an infinite set of processes variables. The class of* CCS processes *is built from the operators of prefix, parallel composition, sum, inaction, recursion as follows:*

$$(Proc_{CCS} \ni)\ p \ ::=\ x \mid a.p \mid p|p \mid p+p \mid 0 \mid rec\ x.p\ ,$$

where $x \in PVar$, and $a \in \mathcal{A}$.
Let Proc^0_{CCS} *denote the subclass of* closed *CCS processes.*

The operational semantics of CCS is given, as usual, by means of a labelled transition system:

Definition 2 (CCS Transition System) *The rules of the transition system for CCS processes are defined as follows:*

$$\frac{}{a.p \xrightarrow{a} p}\ pre \qquad \frac{p \xrightarrow{a} p'}{p+q \xrightarrow{a} p'}\ sum \qquad \frac{p \xrightarrow{a} p'}{p|q \xrightarrow{a} p'}\ par$$

$$\frac{p \xrightarrow{a} p' \quad q \xrightarrow{\bar{a}} q'}{p|q \xrightarrow{a} p'|q'}\ com \qquad \frac{p[recx.p/x] \xrightarrow{a} p'}{recx.p \xrightarrow{a} p'}\ rec$$

(The symmetric rules of parallel composition and summation are omitted.)

In order to define the contextual closure operator, we first need to introduce CCS contexts:

Definition 3 (CCS Contexts) *The class of* CCS contexts *is defined as follows:*

$$C[\ ] \ ::= \ p \ | \ [\ ] \ | \ a.C[\ ] \ | \ C[\ ]|C[\ ] \ | \ C[\ ]+C[\ ] \ | \ rec\ x.C[\ ]\ ,$$

where $p \in Proc_{CCS}$.

Now we are ready to define the contextual closure operator on relations between CCS processes:

Definition 4 (CCS Contextual Closure Operator) *Let* $T_C : \mathcal{P}(Proc^0_{CCS} \times Proc^0_{CCS}) \to \mathcal{P}(Proc^0_{CCS} \times Proc^0_{CCS})$ *be the operator defined as follows:*

$$T_C(\mathcal{R}) = C[\mathcal{R}] = \{(C[p], C[q]) \mid (p,q) \in \mathcal{R} \ \wedge C[\ ] \text{ is a closed CCS context}\}\ .$$

The proof principle "up-to-T_C" allows for smooth proofs of important properties of CCS processes, e.g. unicity of "guarded" fixpoints (see [San98] for more details). Coinduction principles "up-to-context" have been studied also for π-calculus ([MPW92, San98]), and in the context of λ-calculus, for reasoning on various *observational (contextual) equivalences* (see [How96, MST96, Pit96a, Las96, Len97, Len97a, HL99, Las99]). On λ-calculus, *applicative equivalences* have an immediate coinductive characterization in terms of *applicative bisimulations*. Proving the soundness of coinduction principles "up-to-context" for the applicative equivalence is a crucial issue, since it implies that the observational equivalence coincides with the corresponding applicative equivalence, and therefore applicative coinduction and applicative coinduction "up-to-context" can be used for reasoning directly on the observational equivalence.

1.2 Coinduction "up-to-$(\approx, \bullet)$"

Another possible generalization of the Coinduction Principle, in the spirit of Milner's bisimulation "up-to", is based on the following theorem.

Theorem 5 (Coinduction "up-to-$(\approx, \bullet)$") *Let* $\Phi : \mathcal{P}(X \times X) \to \mathcal{P}(X \times X)$ *be a monotone operator, and let* $\bullet : \mathcal{P}(X \times X) \times \mathcal{P}(X \times X) \to \mathcal{P}(X \times X)$ *be an associative operation. If*
1) for all $\mathcal{R}, \mathcal{R}', \mathcal{R}_1, \mathcal{R}'_1$,

$$\mathcal{R} \subseteq \Phi(\mathcal{R}_1) \ \wedge \ \mathcal{R}' \subseteq \Phi(\mathcal{R}'_1) \implies \mathcal{R} \bullet \mathcal{R}' \subseteq \Phi(\mathcal{R}_1 \bullet \mathcal{R}'_1),$$

2) $\approx_\Phi \subseteq \approx_\Phi \bullet \approx_\Phi$,
then
i)

$$\approx_\Phi = \bigcup \{\mathcal{R} | \mathcal{R} \subseteq \Phi(\approx_\Phi \bullet \mathcal{R} \bullet \approx_\Phi)\} \ .$$

ii) The following principle is sound and complete:

$$\frac{x \ \mathcal{R} \ y \qquad \mathcal{R} \subseteq \Phi(\approx_\Phi \bullet \mathcal{R} \bullet \approx_\Phi)}{x \approx_\Phi y} \ .$$

Proof. i) It is sufficient to prove that

a) $\mathcal{R} \subseteq \Phi(\approx_\Phi \bullet \mathcal{R} \bullet \approx_\Phi) \implies \exists \, \mathcal{S} \, . \ \mathcal{R} \subseteq \mathcal{S} \subseteq \Phi(\mathcal{S})$.
b) $\mathcal{R} \subseteq \Phi(\mathcal{R}) \implies \exists \, \mathcal{S} \, . \ \mathcal{R} \subseteq \mathcal{S} \subseteq \Phi(\approx_\Phi \bullet \mathcal{R} \bullet \approx_\Phi)$.

Proof of item a): We prove that $\Phi(\approx_\Phi \bullet \mathcal{R} \bullet \approx_\Phi) \subseteq \Phi(\Phi(\approx_\Phi \bullet \mathcal{R} \bullet \approx_\Phi))$. Then we can take $\mathcal{S} = \Phi(\approx_\Phi \bullet \mathcal{R} \bullet \approx_\Phi)$. From $\mathcal{R} \subseteq \Phi(\approx_\Phi \bullet \mathcal{R} \bullet \approx_\Phi)$ and $\approx_\Phi \subseteq \Phi(\approx_\Phi)$, by item 1, $\approx_\Phi \bullet \mathcal{R} \bullet \approx_\Phi \subseteq \Phi(\approx_\Phi \bullet \approx_\Phi \bullet \mathcal{R} \bullet \approx_\Phi \bullet \approx_\Phi)$. From $\approx_\Phi \subseteq \Phi(\approx_\Phi)$, using item 1, $\approx_\Phi \bullet \approx_\Phi \subseteq \approx_\Phi$. Hence, by item 2, $\approx_\Phi \bullet \approx_\Phi = \approx_\Phi$ and $\approx_\Phi \bullet \mathcal{R} \bullet \approx_\Phi \subseteq \Phi(\approx_\Phi \bullet \mathcal{R} \bullet \approx_\Phi)$. Finally, by monotonicity of Φ, $\Phi(\approx_\Phi \bullet \mathcal{R} \bullet \approx_\Phi) \subseteq \Phi(\Phi(\approx_\Phi \bullet \mathcal{R} \bullet \approx_\Phi))$.
Proof of item b): Since, by the proof of item a), $\approx_\Phi \bullet \approx_\Phi = \approx_\Phi$, then $\approx_\Phi \subseteq \Phi(\approx_\Phi) \implies \approx_\Phi \subseteq \Phi(\approx_\Phi \bullet \approx_\Phi \bullet \approx_\Phi)$. Hence take $\mathcal{S} = \approx_\Phi$.
ii) Immediate consequence of item i of this theorem. $\square$

Milner's bisimulation "up-to" principle is recovered by simply taking as X the domain of CCS processes, as Φ the operator corresponding to strong bisimulation, and as $\bullet$ relational composition.

Hypothesis 1 in Theorem 5 can be viewed as a *generalized transitivity*. In fact, if $\bullet$ is relational composition, then hypothesis 1 implies transitivity of the relation $\approx_\Phi$.

If $\bullet$ is relational composition, and $\approx_\Phi$ is reflexive, then hypothesis 2 of Theorem 5 is satisfied.

Dropping hypothesis 2 in Theorem 5, and assuming the monotonicity of $\bullet$ w.r.t. $\subseteq$, i.e., for all $\mathcal{R}, \mathcal{R}_1, \mathcal{R}', \mathcal{R}'_1$, $\mathcal{R} \subseteq \mathcal{R}_1 \ \wedge \ \mathcal{R}' \subseteq \mathcal{R}'_1 \implies \mathcal{R} \bullet \mathcal{R}' \subseteq \mathcal{R}_1 \bullet \mathcal{R}'_1$, we get soundness of the principle of Coinduction "up-to-($\approx, \bullet$)", but we loose completeness.

1.3 Coiterative and Corecursive Functions

We could give a purely set-theoretic treatment of *coiterative* and *corecursive* functions ([Geu92, Gim94]). But we feel that in this respect the coalgebraic setting is the most natural. We only point out that indeed it would be possible to justify definitions by coiteration and corecursion solely by means of maximal fixed points. In fact, the graphs of coiterative functions can be naturally defined as maximal fixed points, since they are bisimulations after all. This approach also highlights the connections between corecursive morphisms and the Coinduction "up-to-$\cup \approx$" (see Section 4.2 for more details).

2 Coalgebraic Description of Coinduction and Coiterative Morphisms

In this section, we present the categorical description of coinduction based on the notion of *coalgebra*, for capturing equivalences induced by *coiterative morphisms* ([Acz88, AM89, RT93, RT94, Rut96, Tur96, TP97, Len98]). In this setting, the categorical counterparts of set-theoretic bisimulations are F*-bisimulations*, i.e. *spans of coalgebra morphisms* ([TP97]). One of the advantages of a categorical description is that we can deal uniformly with coinductively defined objects and *coiterative* morphisms. In fact, the latter arise naturally in a categorical context.

As we will see formally in Section 4, the coinduction principle based on F-bisimulations presented in this section is the categorical counterpart of the Coinduction Principle 1, in the case of equivalences induced by coiterative morphims.

We start by introducing the category of F-coalgebras, for F endofunctor on a category $\mathcal{C}$. F-coalgebras, i.e. pairs (X, α_X), where $\alpha_X : X \to FX$ is an arrow in $\mathcal{C}$, can be endowed with the structure of a category by introducing the notion of F-coalgebra morphism as follows:

Definition 5 *Let $F : \mathcal{C} \to \mathcal{C}$. Then $f : (X, \alpha_X) \to (Y, \alpha_Y)$ is an F*-coalgebra morphism *if $f : X \to Y$ is an arrow of the category $\mathcal{C}$ such that the following diagram commutes*

$$\begin{tikzcd} X \arrow[r, "f"] \arrow[d, "\alpha_X"'] & Y \arrow[d, "\alpha_Y"] \\ F(X) \arrow[r, "F(f)"'] & F(Y) \end{tikzcd}$$

Unique morphisms into final coalgebras are called *coiterative morphisms*:

Definition 6 (Coiteration Scheme) *Let $F : \mathcal{C} \to \mathcal{C}$, let (X, α_X) be an F-coalgebra, and let $(\nu F, \alpha_{\nu F})$ be a final F-coalgebra. The* coiterative morphism *is the unique F-coalgebra morphism $f : (X, \alpha_X) \to (\nu F, \alpha_{\nu F})$.*

Before introducing the notion of F*-bisimulation*, we need to introduce the notion of span.

Definition 7 *A* span *$(\mathcal{R}, r_1, r_2)$ on objects X, Y consists of an object $\mathcal{R}$ in $\mathcal{C}$, and two ordered arrows, $r_1 : \mathcal{R} \to X$ and $r_2 : \mathcal{R} \to Y$.*
Spans on objects X and Y can be ordered as follows:

$$(\mathcal{R}, r_1, r_2) \leq (\mathcal{R}', r_1', r_2') \iff \exists f : \mathcal{R} \to \mathcal{R}'.\ \forall i = 1, 2.\ r_i = f; r_i' \ .$$

The notion of binary relation is expressed, in a general categorical setting, as an equivalence class of monic spans (see [FS90] for more details). As pointed out in [TP97], F-bisimulations on F-coalgebras can be simply taken to be spans in the category of F-coalgebras:

Definition 8 (F-bisimulation, [TP97]) *Let F be an endofunctor on the category $\mathcal{C}$. A span $(\mathcal{R}, r_1, r_2)$ on objects X, Y is an F*-bisimulation *on the F-coalgebras (X, α_X) and*

(Y, α_Y), *if there exists an arrow of* $\mathcal{C}$, $\gamma : \mathcal{R} \to F(\mathcal{R})$, *such that* $((\mathcal{R}, \gamma), r_1, r_2)$ *is a coalgebra span, i.e.*

$$\begin{tikzcd} X \arrow[d, "\alpha_X"'] & \mathcal{R} \arrow[l, "r_1"'] \arrow[r, "r_2"] \arrow[d, "\gamma"] & Y \arrow[d, "\alpha_Y"] \\ F(X) & F(\mathcal{R}) \arrow[l, "F(r_1)"] \arrow[r, "F(r_2)"'] & F(Y) \end{tikzcd}$$

When the two F-coalgebras (X, α_X) and (Y, α_Y) in the definition above coincide, we will simply say that the span is an F-bisimulation on the F-coalgebra (X, α_X).

The following theorem generalizes the fact that, in set-theoretic categories, equivalences induced by coiterative morphisms can be characterized coinductively as the *greatest* F-bisimulations.

Theorem 6 ([TP97]) *Suppose that* $F : \mathcal{C} \to \mathcal{C}$ *has a final* F*-coalgebra* $(\nu F, \alpha_{\nu F})$. *Let* (X, α_X) *be an* F*-coalgebra, and let* $\mathcal{M} : (X, \alpha_X) \to (\nu F, \alpha_{\nu F})$ *be the coiterative morphism. If* F *preserves weak pullbacks, then*
i) for all F*-bisimulations* $(\mathcal{R}, r_1, r_2)$ *on* (X, α_X), $r_1; \mathcal{M} = r_2; \mathcal{M}$;
ii) the kernel pair of $\mathcal{M}$ *is an* F*-bisimulation on* (X, α_X).

2.1 Coalgebraic Coinduction in Set-theoretic Categories

Since in this paper we are interested in giving explicit coinduction principles, and in formalizing the connections between set-theoretic and coalgebraic coinduction, we will focus in particular on the "concrete" setting of *set-theoretic categories*. These are categories whose objects are sets or classes of a possibly non-wellfounded universe of sets (see [Len98] for more details). Examples of set-theoretic categories are:

- $Set(U)$ ($Set^*(U)$): the category whose objects are the (non-)wellfounded sets belonging to a Universe of ZF_0^- (FCU).
- $Class(U)$ ($Class^*(U)$): the category whose objects are the classes of (non-)wellfounded sets belonging to a Universe of ZF_0^- (FCU).
- $\mathcal{HC}_\kappa(U)$ ($(\mathcal{HC}_\kappa)^*(U)$): the category whose objects are the wellfounded (non-wellfounded) sets whose hereditary cardinal is less than κ.
- $Card$ ($CARD$): the category whose objects are the cardinals (including Ord).

Throughout this paper, we will denote with $\mathcal{C}^{\mathcal{S}}$ one of the set-theoretic categories defined above.

In set-theoretic categories, F-bisimulations are usullay defined to be relations (see Definition 9 below). This definition is equivalent to Definition 8, in the sense that the two notions of F-bisimulations characterize the same equivalence (see Theorem 7 below), although in effect they give rise to different coinduction principles.

Definition 9 ([AM89]) *Let $F : \mathcal{C}^{\mathcal{S}} \to \mathcal{C}^{\mathcal{S}}$.* An F-bisimulation on the F-coalgebras (X, α_X) and (Y, α_Y) is a set-theoretic relation $R \subseteq X \times Y$ such that there exists an arrow of $\mathcal{C}$, $\gamma : \mathcal{R} \to F(\mathcal{R})$, making the following diagram commute:

$$\begin{tikzcd} X \arrow[d, "\alpha_X"'] & \mathcal{R} \arrow[l, "\pi_1"'] \arrow[r, "\pi_2"] \arrow[d, "\gamma"] & Y \arrow[d, "\alpha_Y"] \\ F(X) & F(\mathcal{R}) \arrow[l, "F(\pi_1)"] \arrow[r, "F(\pi_2)"'] & F(Y) \end{tikzcd}$$

Notice that the following theorem, which specializes Theorem 6 to set-theoretic categories, holds for both notions of F-bisimulation (either that of Definition 8 or that of Definition 9). The notation "$x\mathcal{R}y$", for a bisimulation $(\mathcal{R}, r_1, r_2)$ on (X, α_X) and (Y, α_Y) as in Definition 8, stands for $\exists u \in \mathcal{R}.\ < r_1, r_2 > (u) = (x, y)$.

Theorem 7 (Coalgebraic Coinduction) *Suppose that $F : \mathcal{C}^{\mathcal{S}} \to \mathcal{C}^{\mathcal{S}}$ has final F-coalgebra $(\nu F, \alpha_{\nu F})$. Let (X, α_X) be an F-coalgebra. If F preserves weak pullbacks, then the following principle is sound and complete*

$$\frac{x\ \mathcal{R}\ y \quad \mathcal{R}\ \text{ is an } F\text{-bisimulation on } (X, \alpha_X)}{x \sim^F_{(X,\alpha_X)} y},$$

where $\sim^F_{(X,\alpha_X)}$ is the equivalence induced by the coiterative morphism $\mathcal{M} : (X, \alpha_X) \to (\nu F, \alpha_{\nu F})$.

3 T-coiteration

The coiteration scheme introduced in Definition 2 is not powerful enough to capture many interesting functions into final coalgebras. E.g., let $h_0 : S_N \to S_N$ be the function which, given a stream of natural numbers s, yields the stream obtained by replacing the first element of s by the constant 0. One can easily check that the function h_0 cannot be defined using the pure coiteration scheme. More general forms of coiteration schemes are therefore required to overcome this limitation. A typical example is the *corecursion scheme* (see [Geu92]).

In this section, we study, from a categorical standpoint, a significant class of coiteration schemes. In particular, we introduce a suitable categorical generalization of the coiteration scheme, which we call *T-coiteration scheme*. In Section 3.1, we discuss some relevant examples of T-coiteration. In Section 3.2, we introduce the principle of Coalgebraic Coinduction "up-to-T". In Section 3.3, we use it to derive a *sound* proof principle for establishing equivalences induced by T-coiterative morphisms. As we will see in Section 4, the principle of Coalgebraic Coinduction "up-to-T" can be viewed as the categorical counterpart of the set-theoretic Coinduction "up-to-T" of Section 1.

We start by recalling the definition of *pointed endofunctor*, which is the categorical generalization of pointed set:

Definition 10 (Pointed Endofunctor) *A* pointed endofunctor *over a category $\mathcal{C}$ is a pair $< T, \eta >$, where T is an endofunctor on $\mathcal{C}$ and $\eta : Id \dot{\to} T$ is a natural transformation.*

Now we are ready to present the generalized coiteration scheme:

Definition 11 (T-coiteration Scheme) *Let $F : \mathcal{C} \to \mathcal{C}$ be such that F has final coalgebra $(\nu F, \alpha_{\nu F})$, and let $< T, \eta >$ be a pointed endofunctor over $\mathcal{C}$. Then, for any F-coalgebra (TX, α), we can define the T-*coiterative *morphism $h : X \to \nu F$ as $f \circ \eta_X$, where f is the unique F-coalgebra morphism from (TX, α) to the final coalgebra $(\nu F, \alpha_{\nu F})$, i.e.*

$$\begin{tikzcd} X \arrow[r, "\eta_X"] & TX \arrow[r, "f"] \arrow[d, "\alpha"'] & \nu F \arrow[d, "\alpha_{\nu F}"] \\ & FTX \arrow[r, "Ff"'] & F(\nu F) \end{tikzcd}$$

3.1 Examples of T-coiteration

We discuss three special cases of T-coiteration. The first is *corecursion.* In particular we provide an example of a function which is corecursive, but not coiterative. The second special case of T-coiteration which we discuss is *context-coiteration.* As we will see in Section 3.2, this scheme is connected, as expected, to the principle of coinduction "up-to-context". Finally, we present a monad whose corresponding coiteration scheme captures, in a natural way, the weak semantics on CCS processes.

3.1.1 Corecursion

The *corecursion scheme* can be recovered in categories with coproducts, by considering the following monad:

Definition 12 (Corecursion Monad) *Let $\mathcal{C}$ be a category with coproducts, and let $F : \mathcal{C} \to \mathcal{C}$ have final coalgebra $(\nu F, \alpha_{\nu F})$. Let $T_F^+ : \mathcal{C} \to \mathcal{C}$ be the functor defined by*

$$T_F^+ X = X + \nu F \ .$$

The definition of T_F^+ on arrow is canonical, i.e., for all morphisms $f : X \to Y$ in $\mathcal{C}$, $T_F^+ f = [f; in_1, in_2]$.

The functor T_F^+ can be endowed with a structure of monad by defining

- $\eta_X = in_1 : X \to X + \nu_F$
- $\mu_X = [id, in_2] : (X + \nu F) + \nu F \to X + \nu F,$

where in_1, in_2 are the canonical sum injections.

Then, by taking T in Definition 11 to be the monad induced by T_F^+, and by considering F-coalgebras of the shape $(X + \nu F, [\alpha_1, F(in_2)])$, we recover exactly the corecursion scheme. The essence of the corecursion scheme is that we can make a choice between the two possibilities offered by the two branches of the disjoint sum in the F-coalgebra. Notice how this is exploited in the definition of the F_S-coalgebra in Proposition 1.

Proposition 1 *The function $h_0 : S_N \to S_N$, defined by $h_0(n,s) = (0,s)$, which, given a stream of natural numbers s, yields the stream obtained by replacing the first element of s by the constant 0, is the corecursive function induced by:*

- *the functor $F_S : \mathcal{C}^{\mathcal{S}} \to \mathcal{C}^{\mathcal{S}}$ defined on objects by $F_S(X) = N \times X$ (its definition on arrows is canonical), where N denotes the set of natural numbers, and S_N is the set of all infinite streams on natural numbers, which is final coalgebra for F_S;*
- *the F_S-coalgebra $(S_N + S_N, [\alpha_1, F_S(in_2)])$, where $\alpha_1 : S_N \to N \times (S_N + S_N)$ is defined by $\alpha_1(n,s) = (0, in_2(s))$.*

3.1.2 Context-coiteration

Another interesting T-coiteration scheme is the one induced by the following *contextual closure monad* on CCS.

Definition 13 (Contextual Closure Monad) *Let $T_C : \mathcal{C}^{\mathcal{S}} \to \mathcal{C}^{\mathcal{S}}$ be the endofunctor defined as follows:*

$$T_C(X) = C[X] = \{C[x] \mid x \in X \ \wedge \ C[\ ] \textit{ is a closed CCS context}\} \ ,$$

for any morphism $f : X \to Y$ in $\mathcal{C}^{\mathcal{S}}$,

$$T_C(f) = C[x] \mapsto C[f(x)] \ .$$

The functor T_C can be endowed with a structure of monad by defining

- *$\eta_X : X \to T_C(X)$ by $x \mapsto [x]$*
- *$\mu_X : T_C^2(X) \to T_C(X)$ by $C_1[C_2[x]] \mapsto C_1[C_2[\ ]/[\ ]][x]$.*

3.1.3 $\_ + \mathbf{Proc}^0_{CCS}$-coiteration and Weak Semantics

We show how a *weak congruence semantics* on CCS processes can be naturally defined as a T-coiterative function, for a suitable monad T.

We start by recalling the notions of *weak bisimulation* and *weak congruence* ([Mil83]):

Definition 14 (Weak Bisimulation) *A symmetric relation $\mathcal{R} \subseteq Proc^0_{CCS} \times Proc^0_{CCS}$ is a* weak bisimulation *if, for all p, q,*

$$p\mathcal{R}q \implies \forall p_1,\ \forall a.\ (p \xrightarrow{a} p_1 \implies \exists q_1.\ q \stackrel{a}{\Rightarrow} q_1 \ \wedge \ p_1 \mathcal{R} q_1) \ ,$$

where $\stackrel{a}{\Rightarrow}$ denotes $\Rightarrow$ (i.e. the reflexive and transitive closure of $\xrightarrow{\tau}$), if $a = \tau$, and $\Rightarrow \circ \xrightarrow{a} \circ \Rightarrow$, otherwise.
The greatest weak bisimulation, $\approx_w$, is called weak equivalence.

Definition 15 (Weak Congruence) *The relation $\approx_{wc} \subseteq Proc^0_{CCS} \times Proc^0_{CCS}$ is defined by*

$$p \approx_{wc} q \ \textit{ if and only if } \forall p_1,\ a.\ (p \xrightarrow{a} p_1 \implies \exists q_1.\ q \xrightarrow{a} q_1 \ \wedge \ p_1 \approx_w p_1') \ \wedge \ \textit{vice versa} \ .$$

Notice that it is not immediate to give a final semantics (i.e. a coiterative morphism) inducing the weak congruence, since this congruence is not directly defined by coinduction. In principle, it is possible to give an alternative coinductive characterization of the weak congruence, which suggests a possible definition of a final semantics (see [Acz93, Len98]). But this is not immediate, and it is an indirect way of giving a weak congruence semantics. On the contrary, the original definition of weak congruence naturally suggests how to provide a weak congruence semantics by $\_ + \mathrm{Proc}^0_{CCS}$-coiteration, where $\_ + \mathrm{Proc}^0_{CCS}$ is the monad defined as follows:

Definition 16 *Let $T_{Proc^0_{CCS}} : \mathcal{C}^{\mathcal{S}} \to \mathcal{C}^{\mathcal{S}}$ be the functor defined by*

$$T_{Proc^0_{CCS}}(X) = X + Proc^0_{CCS} \ .$$

The definition of $T_{Proc^0_{CCS}}$ on arrow is canonical, i.e., for all morphisms $f : X \to Y$ in $\mathcal{C}$, $T_{Proc^0_{CCS}} f = [f; in_1, in_2]$.

The functor $T_{Proc^0_{CCS}}$ can be endowed with a structure of monad by defining

- $\eta_X = in_1 : X \to X + Proc^0_{CCS}$
- $\mu_X = [id, in_2] : (X + Proc^0_{CCS}) + Proc^0_{CCS} \to X + Proc^0_{CCS}$.

Proposition 2 *Let $F_{CCS} : \mathcal{C}^{\mathcal{S}} \to \mathcal{C}^{\mathcal{S}}$ be the functor defined by*

$$F_{CCS}(X) = \mathcal{P}(\mathcal{A} \times X)$$

with canonical definition on arrows.
Let $(Proc^0_{CCS}, \alpha_w)$ be the F_{CCS}-coalgebra defined by

$$\alpha_w(u) = \begin{cases} \{(a, in_2(p_1)) \mid p \xrightarrow{a} p_1\} & \text{if } u = in_1(p) \\ \{(a, in_2(p_1)) \mid p \overset{a}{\Rightarrow} p_1\} & \text{if } u = in_2(p). \end{cases}$$

Then the equivalence determined by the $T_{Proc^0_{CCS}}$-coiterative function $h_w : Proc^0_{CCS} \to P$, where (P, α_P) is the final F_{CCS}-coalgebra, is the weak congruence.

3.2 Coalgebraic Coinduction "up-to-T"

We present now the categorical version of the set-theoretical principle of Coinduction "up-to-T", for equivalences induced by coiterative morphisms. We call this principle *Coalgebraic Coinduction "up-to-T"*. We will show that the Coalgebraic Coinduction "up-to-T" is related to the *T-coiteration scheme* introduced above, in the sense that it can be used to derive a proof principle for establishing equivalences induced by T-coiterative morphisms.

We start by introducing the notion of F-bisimulation "up-to-T". In order to give this definition, we assume that T is a monad, and not just a pointed endofunctor.

Definition 17 (F-bisimulation "up-to-T") *Let $F : \mathcal{C} \to \mathcal{C}$, let $< T, \eta, \mu >$ be a monad on $\mathcal{C}$, and let (TX, α) and (TY, β) be F-coalgebras. An F-bisimulation "up-to-T" on the the F-coalgebras (TX, α) and (TY, β) is a span $(\mathcal{R}, r_1, r_2)$ on TX and TY, such that there exists an arrow of $\mathcal{C}$, $\gamma : \mathcal{R} \to FT(\mathcal{R})$, making the following diagram commute:*

$$
\begin{tikzcd}
TX \arrow[d, "\alpha"'] & \mathcal{R} \arrow[l, "r_1"'] \arrow[r, "r_2"] \arrow[d, "\gamma"] & TY \arrow[d, "\beta"] \\
FTX & FT(\mathcal{R}) \arrow[l, "Fr_1^\sharp"] \arrow[r, "Fr_2^\sharp"'] & FTY
\end{tikzcd}
$$

where $r_1^\sharp, r_2^\sharp$ are the unique extensions of r_1, r_2 given by the universality property of η in the adjunction between the Eilenberg-Moore category of T-algebras and the category $\mathcal{C}$, i.e. $\eta_{\mathcal{R}}; r_i^\sharp = r_i$.

The following definition will be useful in Theorem 8 below.

Definition 18 *Let $F : \mathcal{C} \to \mathcal{C}$ be a functor, and let $< T, \eta >$ be a pointed endofunctor over $\mathcal{C}$. We say that T* distributes over F w.r.t. η *if there exists a natural transformation $\lambda : TF \dot{\to} FT$ for which the following diagram commutes*

$$
\begin{tikzcd}
& F \arrow[dl, "\eta F"'] \arrow[dr, "F\eta"] & \\
TF \arrow[rr, "\lambda"'] & & FT
\end{tikzcd}
$$

The relation between F-bisimulations "up-to-T" and F-bisimulations is illustrated by the following theorem:

Theorem 8 *Let $\mathcal{C}$ be a category closed under ω-colimits, let $F : \mathcal{C} \to \mathcal{C}$, let $< T, \eta, \mu >$ be a monad over $\mathcal{C}$. If*
1) F, T preserve ω-colimits,
2) T distributes over F w.r.t. η,
3) for all $((\mathcal{R}, \gamma), r_1, r_2)$ F-bisimulation "up-to-T" on F-coalgebras (TX, α) and (TY, β), also $((T(\mathcal{R}), T\gamma; \lambda), r_1^\natural, r_2^\natural)$ is an F-bisimulation "up-to-T" on F-coalgebras (TX, α) and (TY, β), then
i) If $((\mathcal{R}, \gamma), r_1, r_2)$ is an F-bisimulation on (TX, α) and (TY, β), then $((\mathcal{R}, \gamma; F(\eta_{\mathcal{R}})), r_1, r_2)$ is an F-bisimulation "up-to-T" on (TX, α) and (TY, β).
ii) For all $(\mathcal{R}, r_1, r_2)$ F-bisimulation "up-to-T" on (TX, α) and (TY, β), there exists $(\widetilde{\mathcal{R}}, \widetilde{r_1}, \widetilde{r_2})$ F-bisimulation on (TX, α) and (TY, β) such that $\mathcal{R} \leq \widetilde{\mathcal{R}}$.

Proof. The proof of item i) is immediate. In order to prove item ii), let $\widehat{\mathcal{R}}$ be the ω-colimit of the ω-diagram $\{T^n(\mathcal{R}) \stackrel{T^n \eta_{\mathcal{R}}}{\to} T^{n+1}(\mathcal{R})\}_{n \geq 0}$, then we take as $\widetilde{\mathcal{R}}$ the ω-colimit $T(\widehat{\mathcal{R}})$ of the diagram $\{T^n(\mathcal{R}) \stackrel{T^n \eta_{\mathcal{R}}}{\to} T^{n+1}(\mathcal{R})\}_{n \geq 1}$. Let $r_1^{\sharp n}$, for $n \geq 1$ be defined by $r_1^{\sharp 1} = r_1^\sharp$, $r_1^{\sharp n+1} = (r_1^{\sharp n})^\sharp$. Notice that TX with $\{r_1^{\sharp n} : T^n(\mathcal{R}) \to TX\}_{n \geq 1}$ is a cocone for the latter diagram, hence we take as $\widetilde{r_1}$ the unique morphism from the colimit $T(\widehat{\mathcal{R}})$ to the cocone TX. Moreover, since F preserves colimits, $FT(\widehat{\mathcal{R}})$ is the colimit of the diagram $\{FT^n(\mathcal{R}) \stackrel{T^n \eta_{\mathcal{R}}}{\to} FT^{n+1}(\mathcal{R})\}_{n \geq 1}$, FTX with $\{Fr_1^{\sharp n} : FT^n(\mathcal{R}) \to FTX\}_{n \geq 1}$ is a cocone for the same diagram, and $F\widetilde{r_1} : FT(\widehat{\mathcal{R}}) \to FTX$ is the morphism given by the universal property of $FT(\widehat{\mathcal{R}})$. Using the naturality of η and the distributivity law, one can check that FTX with $\{\overbrace{T(\dots(T(T}^{n}\ \gamma);\overbrace{\lambda);\lambda\dots);\lambda}^{n};Fr_1^{\sharp n} : T^n(\mathcal{R}) \to FTX\}_{n\geq 1}$ is a cocone for $\{T^n(\mathcal{R}) \stackrel{T^n\eta_{\mathcal{R}}}{\to} T^{n+1}(\mathcal{R})\}_{n\geq 1}$. In a similar way, also $FT(\widehat{\mathcal{R}})$ can be endowed with a structure of cocone for the same diagram. Let $\widetilde{\gamma} : T(\widehat{\mathcal{R}}) \to FT(\widehat{\mathcal{R}})$ be the unique morphism given by the universal property of $T(\widehat{\mathcal{R}})$. Finally, in order to prove that the following diagram commutes

$$\begin{tikzcd}
TX & T(\widehat{\mathcal{R}}) \arrow[l, "\widetilde{r_1}"'] \arrow[d, "\widetilde{\gamma}"] \\
FTX \arrow[from=u, "\alpha"'] & FT(\widehat{\mathcal{R}}) \arrow[l, "\widetilde{Fr_1}^{\sharp}"]
\end{tikzcd}$$

we use the fact that, by hypothesis 3, for all $n \geq 0$, the following diagram commutes

$$\begin{tikzcd}
TX \arrow[d, "\alpha"'] & T^n(\mathcal{R}) \arrow[l, "r_1^{\sharp n}"'] \arrow[d, "\overbrace{T(\dots(T(T}^{n}\gamma);\overbrace{\lambda);\lambda\dots);\lambda}^{n}"] \\
FTX & FT^{n+1}(\mathcal{R}) \arrow[l, "Fr_1^{\sharp n+1}"]
\end{tikzcd}$$

This implies that the cocone FTX, $\{\overbrace{T(\dots(T(T}^{n}\ \gamma);\overbrace{\lambda);\lambda\dots);\lambda}^{n};Fr_1^{\sharp n} : T^n(\mathcal{R}) \to FTX\}_{n\geq 1}$ coincides with the cocone FTX with $\{r_1^{\sharp n};\alpha : T^n(\mathcal{R}) \to FTX\}_{n\geq 1}$, and the thesis follows exploiting the universality of $T(\widehat{\mathcal{R}})$ with respect to this cocone. $\square$

Specializing Theorem 8 to endofunctors on a set-theoretic category $\mathcal{C}^{\mathcal{S}}$, we obtain an alternative characterization of the greatest F-bisimulation $\sim^F_{(TX,\alpha)}$ on the F-coalgebra (TX,α), which yields the following

Theorem 9 (Coalgebraic Coinduction "up-to-T") *Let $F : \mathcal{C}^{\mathcal{S}} \to \mathcal{C}^{\mathcal{S}}$ and let T be a monad over $\mathcal{C}^{\mathcal{S}}$ satisfying the hypothesis 1,2,3 of Theorem 8 above. If moreover F preserves weak pullbacks and has final coalgebra, then the following principle is sound and complete*

$$\frac{x\ \mathcal{R}\ y \quad \mathcal{R}\ \ F\text{-bisimulation "up-to-}T\text{" on } (TX,\alpha)}{x \sim^F_{(TX,\alpha)} y}\ .$$

This principle can be specialized, for instance, to the corecursion monad $T^+_{F_S} : \mathcal{C}^{\mathcal{S}} \to \mathcal{C}^{\mathcal{S}}$, where F_S is the functor defined in Proposition 1, and to the CCS contextual closure monad T_C together with the functor F_{CCS}, thus obtaining a coalgebraic principle of bisimulation "up-to-context" for CCS processes. The principle of Coalgebraic Coinduction "up-to-T" of Theorem 9 is related to the set-theoretic Coinduction "up-to-T" of Theorem 4, in the sense made precise by Proposition 4 of Section 4.

3.3 Reasoning on Equivalences of T-coiterative Morphisms

In this section, the principle of Coalgebraic Coinduction "up-to-T" is put to use for reasoning on equivalences induced by T-coiterative morphisms.

First, we need to recall the notions of *image* and *inverse image* of spans. These are to be intended as the (inverse) image of the subobject of $X_1 \times X_2$ determined by the relation underlying a span on X_1, X_2. See [FS90] for more details.

Definition 19 ((Inverse) Image of Spans) *Let $\mathcal{C}$ be a category with products and pull-packs.*

- *The* image *of a span $(\mathcal{R}, r_1, r_2)$ on X_1, X_2 by $(f_1 : X_1 \to Y_1, f_2 : X_2 \to Y_2)$, denoted by $(f_1, f_2)^+(\mathcal{R}, r_1, r_2)$, is the span $(\mathcal{R}, r_1; f_1, r_2; f_2)$ on Y_1, Y_2.*
- *The* inverse image *of a span $(\mathcal{R}, r_1, r_2)$ on X_1, X_2 by $(f_1 : X_1 \to Y_1, f_2 : X_2 \to Y_2)$, denoted by $(f_1, f_2)^{-1}(\mathcal{R}, r_1, r_2)$, is the span $(\mathcal{P}, p_2; \pi_1, p_2; \pi_2)$ on Y_1, Y_2, where $(\mathcal{P}, p_1, p_2)$ is the pullback of $< r_1, r_2 >: \mathcal{R} \to X_1 \times X_2$ and $< \pi_1; f_1, \pi_2; f_2 >: Y_1 \times Y_2 \to X_1 \times X_2$.*

If $(\mathcal{R}, r_1, r_2)$ is a span on X and $f : X \to Y$, $g : Y \to X$, we simply denote by $f^+(\mathcal{R}, r_1, r_2)$ the image of $(\mathcal{R}, r_1, r_2)$ by (f, f), and we denote by $g^{-1}(\mathcal{R}, r_1, r_2)$ the inverse image of $(\mathcal{R}, r_1, r_2)$ by (g, g).

Using the principle of Coalgebraic Coinduction "up-to-T", we now prove the following theorem

Theorem 10 (Coalgebraic Coinduction for T-coiterative Functions) *Let $F : \mathcal{C}^{\mathcal{S}} \to \mathcal{C}^{\mathcal{S}}$ and let $< T, \eta, \mu >$ be a monad on $\mathcal{C}^{\mathcal{S}}$ satisfying all the hypotheses of Theorem 9. Let h be the T-coiterative morphism induced by the F-coalgebra (TX, α), i.e. $h = f \circ \eta_X$, where $f : TX \to \nu F$ is the coiterative morphism. Then the following principle is sound*

$$\frac{x\ \mathcal{R}\ y \quad \eta_X^+(\mathcal{R}, r_1, r_2)\ F\text{-}bisimulation\ \text{“up-to-}T\text{''}\ on\ (TX, \alpha)}{x \sim_h y} ,$$

where $\sim_h$ denotes the equivalence induced by the T-coiterative morphism h.

Proof. One can easily check, using Theorem 9, that

$$\eta_X^+(\mathcal{R}, r_1, r_2)\ F\text{-bisimulation “up-to-}T\text{'' on } (TX, \alpha) \Longrightarrow R \leq \eta_X^{-1}(\sim^F_{(TX,\alpha)}, \pi_1, \pi_2) .$$

The completeness of the categorical coinduction principle for T-coiterative functions requires further conditions. We shall address this issue in another paper.

4 From Coalgebras to Sets and back

In this section we study and discuss the relations between set-theoretic and coalgebraic accounts of coinduction. As we pointed out in the Introduction, this area is quite unexplored and problematic. Here we present some results and raise some problems.

As far as the direction "From Coalgebras to Sets", in the case of functors which preserve weak pullbacks, one can show how to generate, from the coalgebraic coinduction principle based on F-bisimulations of Theorem 7, a corresponding set-theoretic Coinduction Principle 1. For a special class of covariant functors, we show that this translation is *compositional*, in a sense to be made precise. We generalize the translation from coalgebras to sets to coinduction "up-to", bringing to light the connections between the Coalgebraic Coinduction "up-to-T" and the set-theoretic Coinduction "up-to-T".

We work in set-theoretic categories. It would be interesting to extend these results to other possibly more general categorical settings, where also contravariant and mixed functors could be used.

It would be extremely interesting to be able to provide coalgebraic coinduction principles in all contexts where set-theoretic coinduction principles of some kind are at work, but it appears very difficult, even if we restrict ourself to the case of binary relations. On one hand F-bisimulations convey more information than set-theoretic bisimulations. On the other hand it might not be always the case that one *can* give categorical descriptions at all of set-theoretic coinduction, see the examples in Section 4.2.

4.1 From Coalgebras to Sets

We start by introducing some notation. Let us $(\mathcal{R}, r_1, r_2)$ be a span on X and Y, for X, Y objects of a set-theoretic category $\mathcal{C}^{\mathcal{S}}$. Let denote by $\mathcal{R}^{\mathcal{S}}_{r_1 r_2}$ the set-theoretic relation induced by $(\mathcal{R}, r_1, r_2)$, i.e.

$$\mathcal{R}^{\mathcal{S}}_{r_1 r_2} = \{(x,y) \in X \times Y \mid \exists u \in \mathcal{R} \ . \ < r_1, r_2 > (u) = (x,y)\} \ .$$

Rutten, in [Rut98], using the theory of *relators*, showed that, when $F : Set \to Set$ preserves weak pullbacks, $\mathcal{R}$ is an F-bisimulation if and only if $\mathcal{R}$ is an $\mathcal{F}$-coalgebra morphism, where $\mathcal{F} : Rel \to Rel$ is the relator extending F. In purely set-theoretical terms, Corollary 3.1 of [Rut98] can be spelled out as follows:

Proposition 3 *Let $F : \mathcal{C}^{\mathcal{S}} \to \mathcal{C}^{\mathcal{S}}$, let $(\mathcal{R}, r_1, r_2)$ be a span on X, Y, and let (X, α), (Y, β) be F-coalgebras. Then*
i) $(\mathcal{R}, r_1, r_2)$ is an F-bisimulation on (X,α) and (Y,β) $\Longleftrightarrow$ $\mathcal{R}^{\mathcal{S}}_{r_1 r_2} \subseteq \Phi^F_{(X,\alpha),(Y,\beta)}(\mathcal{R}^{\mathcal{S}}_{r_1 r_2})$,
where $\Phi^F_{(X,\alpha),(Y,\beta)} : \mathcal{P}(X \times Y) \to \mathcal{P}(X \times Y)$ is defined by

$$\Phi^F_{(X,\alpha),(Y,\beta)}(\mathcal{R}) = \{(x,y) \mid \exists u \in F(\mathcal{R}). \ (F(\pi_1^{\mathcal{R}})(z) = \alpha(x) \ \wedge \ F(\pi_2^{\mathcal{R}})(z) = \beta(y))\} \ .$$

ii) Moreover, if F preserves weak pullbacks, then $\Phi^F_{(X,\alpha),(Y,\beta)}$ is monotone.

Now we show that, for a special class $\mathcal{F}un$ of functors, the translation from the categorical coinduction to the set-theoretical coinduction of Proposition 3 is *compositional* on the structure of $F \in \mathcal{F}un$. I.e., given $F \in \mathcal{F}un$ and F-coalgebras $(X,\alpha), (Y,\beta)$, there is a natural way of inducing coalgebras of the "component" functors of F, in such a way that the operator $\Phi^F_{(X,\alpha),(Y,\beta)}$ is obtained by "composing" the operators induced by the component functors. This is related to the work of [HJ98] in the fibrational setting.

We start by specifying a class $\mathcal{F}un$ of *covariant* functors. The functors which we consider involve the constructors which are normally used for defining final semantics, i.e. identity, constants, cartesian and infinite cartesian products, disjoint sum, powerset constructors.

Definition 20 *Let $\mathcal{F}un$ be the class of functors $F : \mathcal{C}^{\mathcal{S}} \to \mathcal{C}^{\mathcal{S}}$ defined as follows:*

$$F(\cdot) ::= Id(\cdot) \;\mid\; F_C(\cdot) \;\mid\; F(\cdot) \times F(\cdot) \;\mid\; F(\cdot) + F(\cdot) \;\mid\; \mathcal{P}(F(\cdot)) \;\mid\; F_C(\cdot) \to F(\cdot) \;,$$

where

- *$Id(\cdot)$ is the identity functor, defined by*

$$\begin{cases} Id(A) = A & \text{for } A \text{ object in } \mathcal{C}^{\mathcal{S}} \\ Id(f) = f & \text{for } f \text{ arrow in } \mathcal{C}^{\mathcal{S}} \end{cases},$$

- *$F_C(\cdot)$, for C object in $\mathcal{C}^{\mathcal{S}}$, is the constant functor, defined by*

$$\begin{cases} F_C(A) = C & \text{for } A \text{ object in } \mathcal{C}^{\mathcal{S}} \\ F_C(f) = id_C & \text{for } f \text{ arrow in } \mathcal{C}^{\mathcal{S}} \end{cases},$$

Compositionality of the translation given in Proposition 3 can be expressed as follows:

Theorem 11 *Let $F : \mathcal{C}^{\mathcal{S}} \to \mathcal{C}^{\mathcal{S}}$ be a functor in $\mathcal{F}un$, and let (X, α), (Y, β) be F-coalgebras.*

($\times$) *If $F(\cdot) = F_1(\cdot) \times F_2(\cdot)$, then, for all $\mathcal{R}$,*

$$\Phi^F_{(X,\alpha),(Y,\beta)}(\mathcal{R}) = \Phi^{F_1}_{(X,\pi_1\circ\alpha),(Y,\pi_1\circ\beta)}(\mathcal{R}) \;\cap\; \Phi^{F_2}_{(X,\pi_2\circ\alpha),(Y,\pi_2\circ\beta)}(\mathcal{R}) \;.$$

($+$) *If $F(\cdot) = F_1(\cdot) + F_2(\cdot)$, then, for all $\mathcal{R}$,*

$$\Phi^F_{(X,\alpha),(Y,\beta)}(\mathcal{R}) = \Phi^{F_1}_{(X_1,\alpha_1),(Y_1,\beta_1)}(\mathcal{R} \cap (X_1 \times Y_1)) \cup \Phi^{F_2}_{(X_2,\alpha_2),(Y_2,\beta_2)}(\mathcal{R} \cap (X_2 \times Y_2)) \;,$$

where the F_i-coalgebras (X_i, α_i), (Y_i, β_i) are defined as follows. First of all, notice that the F-coalgebras (X, α) and (Y, β) are of the shape: $\alpha = [\alpha'_1, \alpha'_2] : X_1 + X_2 \to F_1(X) + F_2(X)$, with $\alpha'_i : X_i \to F_i(X)$, and $\beta = [\beta'_1, \beta'_2] : Y_1 + Y_2 \to F_1(Y) + F_2(Y)$, with $\beta'_i : Y_i \to F_i(Y)$. Then $\alpha_i : X \to F_i(X)$ is any F_i-coalgebra such that $(\alpha_i)_{|X_i} = \alpha'_i$, and $\beta_i : Y \to F_i(Y)$ is any F_i-coalgebra such that $(\beta_i)_{|Y_i} = \beta'_i$.

($\mathcal{P}$) *If $F(\cdot) = \mathcal{P}(F_1(\cdot))$, then, for all $\mathcal{R}$,*

$$\Phi^F_{(X,\alpha),(Y,\beta)}(\mathcal{R}) = \bigcap_{\alpha_i}\bigcup_{\beta_j} \Phi^{F_1}_{(X,\alpha_i),(Y,\beta_j)}(\mathcal{R}) \;\cap\; \bigcap_{\beta_j}\bigcup_{\alpha_i} \Phi^{F_1}_{(X,\alpha_i),(Y,\beta_j)}(\mathcal{R}) \;,$$

where the F_1-coalgebras (X, α_i), (Y, β_j) are induced by the F-coalgebras (X, α) and (Y, β) as follows. $\alpha_i : X \to F_1(X)$ is such that $\forall x \in X.\ \alpha_i(x) \in \alpha(x)$, and $\beta_j : Y \to F_1(Y)$ is such that $\forall y \in Y.\ \beta_j(y) \in \beta(y)$.

($\to$) *If $F(\cdot) = C \to F_1(\cdot)$, then, for all $\mathcal{R}$,*

$$\Phi^F_{(X,\alpha),(Y,\beta)}(\mathcal{R}) = \bigcap_{c \in C} \Phi^{F_1}_{(X,\alpha_c),(Y,\beta_c)}(\mathcal{R}) \;,$$

where $\alpha_c : X \to F_1(X)$, is $\lambda x.\alpha(x)(c)$, and $\beta_c : Y \to F_1(Y)$, is $\lambda y.\beta(y)(c)$.

We address now the problem of formalizing the correspondence between Coalgebraic Coinduction "up-to-T" and set-theoretic Coinduction "up-to-T".

Proposition 4 *Let $F : \mathcal{C}^{\mathcal{S}} \to \mathcal{C}^{\mathcal{S}}$ be a functor, let $< T, \eta, \mu >$ be a monad on $\mathcal{C}^{\mathcal{S}}$, let $(\mathcal{R}, r_1, r_2)$ be a span on objects TX, TY, and let (TX, α), (TY, β) be F-coalgebras. Then $(\mathcal{R}, r_1, r_2)$ is an F-bisimulation "up-to-T" on (TX, α) and (TY, β) $\Longleftrightarrow$*

$$\mathcal{R}^{\mathcal{S}}_{r_1 r_2} \subseteq \Phi^F_{(TX,\alpha),(TY,\beta)}(\Phi^T_{X,Y}(\mathcal{R}^{\mathcal{S}}_{r_1 r_2})) \ ,$$

where $\Phi^T_{X,Y} : \mathcal{P}(TX \times TY) \to \mathcal{P}(TX \times TY)$ is defined by

$$\Phi^T_{X,Y}(\mathcal{R}) = (TR)^{\mathcal{S}}_{\pi_1^\sharp \pi_2^\sharp} \ ,$$

and $\Phi^F_{(TX,\alpha),(TY,\beta)}$ is the operator defined in Proposition 3.

It would be interesting to find suitable counterparts to Theorem 11 for Coinduction "up-to-T".

4.1.1 Examples

We discuss the translation from coalgebraic bisimulations "up-to" to set-theoretical bisimulations "up-to", given in Proposition 4, in some special cases.

Coinduction "up-to-$T^+_{F_S}$". We apply Proposition 4 to the monad for corecursion $T^+_{F_S}$ together with the functor F_S defined in Section 3.1.1. Let $(T^+_{F_S}(S_N), [\alpha_1, F_S(in_2)])$ be an F_S-coalgebra. Then, by Proposition 4, using strong extensionality of final coalgebras, one can easily check that a relation $(\mathcal{R}, r_1, r_2)$ is an F_S-bisimulation "up-to-$T^+_{F_S}$" on $(T^+_{F_S}(S_N), [\alpha_1, F_S(in_2)])$ if and only if $\mathcal{R}^{\mathcal{S}}_{r_1 r_2}$ is a Φ^+-bisimulation "up-to-$\cup\approx$" for the operator $\Phi^+ : \mathcal{P}(T^+_{F_S}(S_N) \times T^+_{F_S}(S_N)) \to \mathcal{P}(T^+_{F_S}(S_N) \times T^+_{F_S}(S_N))$ defined by

$$\begin{aligned}\Phi^+(\mathcal{R}) = \{&(in_1(s), in_1(s')) \mid \pi_1(\alpha_1(s)) = \pi_1(\alpha_1(s')) \ \wedge \ \pi_2(\alpha_1(s)) \ \mathcal{R} \ \pi_2(\alpha_1(s'))\} \ \cup \\ \{&(in_2(s), in_2(s')) \mid s = s'\}.\end{aligned}$$

The correspondence between Coalgebraic Coinduction "up-to-T^+_F" and set-theoretic Coinduction "up-to-$\cup\approx$" is quite intrinsic. Categorically, applying Theorem 10 to the corecursion monad T^+_F, we get a proof principle for reasoning on equivalences of corecursive morphisms. While, in a purely set-theoretic framework, one can give a coinductive characterization of the equivalence induced by corecursive functions, using the Coinduction "up-to-$\cup\approx$". For simplicitly, we work out only the special case of the functor F_S introduced in Subsection 3.2.

Notice that in the set-theoretic case we derive a *complete* characterization. This immediately implies, by Proposition 4, also the completeness of the Coalgebraic Coinduction "up-to-$T^+_{F_S}$".

Theorem 12 (Set-theoretic Coinduction for Corecursive Functions) *Let $h : X \to S_N$ be the corecursive morphisms induced by the F_S-coalgebra $(T^+_{F_S}X, [\alpha_1, F_S(in_2)])$, i.e. $h = f \circ in_1$, where $f : T^+_{F_S}X \to S_N$ is the coiterative morphism. Then the following principle is sound and complete:*

$$\frac{x \; \mathcal{R} \; y \quad in_1^+(\mathcal{R}) \subseteq \Phi^+(in_1^+(\mathcal{R}) \cup \approx_{\Phi_f})}{x \sim_h y} \quad ,$$

where $in_1^+(\mathcal{R})$ denotes the set-theoretic image of $\mathcal{R}$ by in_1, and Φ_f is the monotone operator $\Phi^{F_S}_{(T^+_{F_S}X,[\alpha_1,F(in_2)]),(T^+_{F_S}X,[\alpha_1,F(in_2)])}$ given in Proposition 3.

Proof. First of all notice that, using Theorem 7 of Section 2,
$\sim_h = in_1^{-1}(\sim_f) = \bigcup\{in_1^{-1}(\mathcal{R}) \mid \mathcal{R} \subseteq (X + S_N)^2 \ \wedge \ \mathcal{R} \subseteq \Phi_f(\mathcal{R})\}$.
We prove that
$\bigcup\{in_1^{-1}(\mathcal{R}) \mid \mathcal{R} \subseteq (X + S_N)^2 \ \wedge \ \mathcal{R} \subseteq \Phi_f(\mathcal{R})\} \ =$
$\bigcup\{\mathcal{R} \mid \mathcal{R} \subseteq X \times X \ \wedge \ in_1^+(\mathcal{R}) \subseteq \Phi_f(in_1^+(\mathcal{R}) \cup \approx_{\Phi_f})\}$.
($\subseteq$) Let $\mathcal{R} \subseteq \Phi_f(\mathcal{R})$.
Then $in_1^+(in_1^{-1}(\mathcal{R})) \subseteq \mathcal{R} \subseteq \Phi_f(\mathcal{R}) \subseteq \Phi_f(\approx_{\Phi_f}) \subseteq \Phi_f(in_1^+(in_1^{-1}(\mathcal{R})) \cup \approx_{\Phi_f})$.
($\supseteq$) Let $\mathcal{R} \subseteq X \times X$ be such that $in_1^+(\mathcal{R}) \subseteq \Phi_f(in_1^+(\mathcal{R}) \cup \approx_{\Phi_f})$. Then, by the Principle "up-to-$\cup \approx_{\Phi_f}$", $in_1^+(\mathcal{R}) \subseteq \approx_{\Phi_f}$. Hence $\mathcal{R} \subseteq in_1^{-1}(\approx_{\Phi_f})$.

Corollary 1 *The Coalgebraic Coinduction "up-to-$T^+_{F_S}$" is complete.*

Coinduction "up-to-context". We consider our second running example monad, i.e. the CCS contextual closure monad T_C. As one can expect, the coalgebraic coinduction principle induced by this monad corresponds to the set-theoretical principle "up-to-context". For instance, Proposition 4 applied to the monad T_C, the functor F_{CCS}, and the F_{CCS}-coalgebra $(T_C(\mathrm{Proc}^0_{CCS}), \alpha_s)$, where α_s is defined by $\alpha_s(C[p]) = \{(a, C[p_1]) \mid p \xrightarrow{a} p_1\}$, yields the following correspondence. A relation $(\mathcal{R}, r_1, r_2)$ is an F_{CCS}-bisimulation "up-to-T_C" on the F_{CCS}-coalgebra $(T_C(\mathrm{Proc}^0_{CCS}), \alpha_s)$ iff $\mathcal{R}^{\mathcal{S}}_{r_1,r_2}$ is a Φ_{CCS}-bisimulation "up-to-context" for the operator $\Phi_s : \mathcal{P}(T_C(Proc^0_{CCS}) \times T_C(Proc^0_{CCS})) \to \mathcal{P}(T_C(Proc^0_{CCS}) \times T_C(Proc^0_{CCS})$ for strong bisimulation defined by

$\Phi_s(\mathcal{R}) = \{(u, v) \mid$
$(\forall (a, u_1) \in \alpha(u).\ \exists (a, v_1) \in \alpha_s(v).\ u_1 \ \mathcal{R} \ v_1) \wedge (\forall (a, v_1) \in \alpha_s(v).\ \exists (a, u_1) \in \alpha_s(u).\ u_1 \ \mathcal{R} \ v_1)\}$.

Notice that Theorem 10 guarantees also the completeness of the principle "up-to-context", while Theorem 3 ensures only the soundness of the set-theoretic coinduction "up-to-context". This is justified by the fact that Coalgebraic Coinduction captures special kinds of equivalences, i.e. those induced by coiterative morphisms. So, for instance Theorem 10 does not subsume the case of the operator costantly equal to the least fixed point of the operator Φ, provided in Section 1.1 as counterexample to the completeness of the principle in Theorem 3.

Coinduction "up-to-$\_ + \mathrm{Proc}^0_{CCS}$". Finally, the $\_ + \mathrm{Proc}^0_{CCS}$-monad, together with the functor F_{CCS} and the F_{CCS}-coalgebra for the weak semantics presented in Section 3.1.3, give rise, set-theoretically, to a principle of coinduction "up-to-$\cup$", where the union is taken with the identity relation on processes in the second branch of the $+$, and the operator $\phi^{F_{CCS}}_{(Proc^0_{CCS}, \alpha_w)}$ is defined as the operator Φ_s, but substituting α_w in place of α_s.

4.2 From Sets to Coalgebras?

In this subsection we list some critical situations where set-theoretic coinductions do not seem to be directly expressible in categorical terms. These examples possibly indicate some limitations of the coalgebraic approach.

4.2.1 Non-uniform Bisimulations.

Consider, for the sake of example, the following notion of bisimulation on CCS like processes, obtained by sligthly modifying the definition of strong bisimulation:

$p \, \mathcal{R} \, q \implies$
either $(p \not\to \ \wedge q \not\to)$
or $\exists a. p \stackrel{a}{\to} \ (\forall p_1 (p \stackrel{a}{\to} p_1 \Rightarrow \exists q_1.\, q \stackrel{a}{\to} q_1 \wedge p_1 \, \mathcal{R} \, q_1) \ \wedge \ \forall q_1 (q \stackrel{a}{\to} q_1 \Rightarrow \exists p_1.\, p \stackrel{a}{\to} p_1 \wedge p_1 \, \mathcal{R} \, q_1))$,

where $p \not\to$ means that there is no transition from p, and $p \stackrel{a}{\to}$ means that there is a transition from p with action a.
It is not at all clear how to describe this notion of bisimulation coalgebraically. The problem is due to the presence of an $\exists$ quantifier, in place of a $\forall$. Intuitively, $\forall$ quantifiers guarantee a uniform property to hold over all objects. With $\exists$ quantifiers we lose this uniformity. But this uniformity seems necessary in providing a coalgebraic description. More in general, the problem with $\exists$ quantifiers can be rephrased as follows.

Let $\Phi_1, \dots, \Phi_n : \mathcal{P}(X \times X) \to \mathcal{P}(X \times X)$ be monotone operators. Each of these operators generates a coinduction principle in the line of the Coinduction Principle 1. If we define $\Phi : \mathcal{P}(X \times X) \to \mathcal{P}(X \times X)$ by

$$\Phi(\mathcal{R}) = \bigcup_{1 \leq i \leq n} \Phi_i(\mathcal{R}) \ ,$$

we get a monotone operator which gives rise to a corresponding coinduction principle. Assuming we have coalgebraic descriptions of the set-theoretic coinduction principles induced by the operators Φ_i, it is not at all clear how to derive a coalgebraic description of the coinduction principle induced by Φ. A similar example occurs in [HL99] for the case of a generalized *applicative* coinduction principle for λ-calculus.

Other examples of bisimulations which have a problematic coalgebraic description are those where "side-conditions" depending on the structure of the objects to be related appear. Both *early* and *late* bisimulations in Milner's π-calculus ([MPW92]), are of this form. Also in this case, like in the previous example with quantifiers, we lack a uniform description. Luckily, in the π-calculus case, it is still possible to get rid of the local side-conditions in the definitions of bisimulations (see [HLMP98]), thereby making possible a coalgebraic description. This latter situation seems related to the difficulty of obtaining a "generalized minimal automata".

4.2.2 Coinduction "up-to".

In this paper, we have discussed coalgebraic counterparts to set-theoretic Coinduction "up-to-T". Not all operators T, however, appear immediately tractable in coalgebraic terms. For example, consider the set-theoretic operator T defined by $\approx \circ \_ \circ \_ \circ \approx$, which captures

Milner's bisimulation "up-to" principle. Similarly, coalgebraic counterparts for the set-theoretic Coinduction "up-to-($\approx, \bullet$)" of Section 1.2 are problematic. The theory of functors and relators could shed some light on this problem.

4.2.3 Binary Operators.

Let $\Phi : \mathcal{P}(X \times Y) \times \mathcal{P}(X \times Y) \to \mathcal{P}(X \times Y) \times \mathcal{P}(X \times Y)$ be a monotone operator over the complete lattice of relations ordered by $(\leq_1, \leq_2)$, where $\leq_i \in \{\subseteq, \supseteq\}$. A special case is that of the *mixed induction-coinduction* operators. These are monotone operators on the complete lattice $\mathcal{P}(X \times Y) \times \mathcal{P}(X \times Y)$ ordered by $(\subseteq, \supseteq)$, which have a unique fixed point $(\approx, \approx)$. These kind of operators induce an *induction-coinduction principle* of the following form (see e.g. [HL95, Pit96]):

$$\frac{\mathcal{R}^- \subseteq \pi_1(\Phi(\mathcal{R}^-, \mathcal{R}^+)) \quad \mathcal{R}^+ \supseteq \pi_2(\Phi(\mathcal{R}^-, \mathcal{R}^+))}{\mathcal{R}^- \subseteq \approx \subseteq \mathcal{R}^+} \ .$$

It is not clear how to describe coalgebraically coinduction principles induced by these binary operators. In particular, induction-coinduction principles seem to require an extension of the coalgebraic approach to contravariant (mixed) functors. Freyd's algebraically compact categories could be of some help here. However, it appears to be problematic to express coalgebraically in full generality already the "purely covariant" case, in which both components of the binary operator are ordered by $\subseteq$. Similar problems, of course, arise for n-ary operators.

5 Final Remarks

In this paper, we have made some contributions in the direction of investigating the correspondence between set-theoretical and coalgebraic descriptions of coinduction, for equivalences induced by coiterative morphisms. In particular, we have discussed principles of bisimulation "up-to", and their connections to generalized coiteration schemes.

There are many problems that still remain to be addressed, some of them we have already mentioned in the previous sections. Here we point at three further important directions of future work.

- It would be useful to study an algebra of monads for which the principle of Coalgebraic Coinduction "up-to" is sound, in the line of the algbera of *respectful functions* developed in [San98]. This would consists in showing that the coalgebraic principle holds for some basic monads, and then showing that the class of monads for which the principle holds is closed under suitable combinations of monads, e.g. composition, iteration, concatenation. One important primitive monad is the contextual closure, which we have discussed in this paper. Another important, basic operation we would like to have, but which is problematic in our context, is the constant to-$\approx$-operation. If we would have this operation and closure under compositionality, we could recover Milner's original bisimulation "up-to" principle as special case in our coalgebraic setting.

- More work is necessary at the categorical level. In the definition of T-coiteration scheme, T is only required to be a pointed endofunctor, while the notion of F-bisimulation "up-to-T" is given for T monad. In principle, this latter condition could be weakened, but in practise the examples considered in this paper involve monads. More work is necessary in order to find suitable conditions on pointed endofunctors to be monads. This would allow to find alternative (possible weaker) conditions for the Coalgebraic Coinduction "up-to-T" to hold.
- In Section 3.1.3, we have seen how a weak semantics on CCS processes can be naturally defined by T-coiteration, for an appropriate T. Similarly, also a *trace semantics* can be easily given by $\mathcal{P}(\ )$-coiteration, where $\mathcal{P}(\ )$ is the non-deterministic choice monad. This semantics has to be compared to that defined in [PT99], where a trace semantics is given by coiteration, by exploiting a distributivity property between an appropriate "behaviour functor" and the monad $\mathcal{P}(\ )$. More generally, the investigation carried out in this paper seems to suggest that a natural way for defining *fully-abstract* semantics is by generalized coiteration. The framework of generalized coiteration and coinduction "up-to" seems to be a more flexible framework than that of standard coiteration and coinduction. But still much work remains to be done in order to develop a general theory, ensuring good properties of the semantics defined by generalized coiteration, e.g. *compositionality*, in the line of [TP97].

References

[Acz88] P.Aczel. *Non-well-founded sets*, CSLI Lecture Notes **14**, Stanford 1988.

[Acz93] P.Aczel. *Final Universes of Processes*, *MFPS'93* Conference Proceedings, Brookes et al. eds., Springer LNCS **802**, New Orleans 1993.

[AM89] P.Aczel, N.Mendler. A Final Coalgebra Theorem, in *Category Theory and Computer Science*, D.H.Pitt et al. eds., Springer LNCS **389**, 1989, 357–365.

[BV96] J.de Bakker, E.de Vink. *Control Flow Semantics*, MIT Press, 1996.

[BM96] J.Barwise, L.Moss. *Vicious circles: On the mathematics of non-wellfounded phenomena*, CSLI Publications, Stanford, 1996.

[Coq94] T.Coquand. Infinite Objects in Type Theory, in *TYPES-93*, SLNCS **806**, 1994, 62–78.

[Fio96] M.Fiore. A Coinduction Principle for Recursive Data Types Based on Bisimulation, *Information and Computation* **127**, 1996, 186–198.

[FH83] M.Forti, F.Honsell. Set theory with free construction principles, *Ann. Scuola Norm. Sup. Pisa*, Cl. Sci. (4) **10**, 1983, 493– 522.

[FHL94] M.Forti, F.Honsell, M.Lenisa. Processes and Hyperuniverses, *MFCS'94* Conference Proceedings, I.Prívara et al. eds., Springer LNCS **841**, Berlin 1994, 352–361.

[FS90] P.Freyd, A.Scedrov. *Categories, Allegories*, North-Holland, Amsterdam, 1990.

[Geu92] H.Geuvers. Inductive and Coinductive types with Iteration and Recursion, Draft, 1992.

[Gim94] E.Giménez. Codifying guarded definitions with recursive schemes, *Workshop on Types for Proofs and Programs*, P.Dybjer et al. eds., Springer LNCS **996**, 1994, 39–59.

[HJ98] C.Hermida, B.Jacobs. Structural induction and coinduction in a fibrational setting, *Inf. and Comp.* **145**, 107–152, 1998.

[HL95] F.Honsell, M.Lenisa. Final Semantics for Untyped Lambda Calculus, *TLCA'95* Conf. Proc., M.Dezani, G.Plotkin eds., Springer LNCS **902**, Berlin 1995, 249–265.

[HL99] F.Honsell, M.Lenisa. Coinductive Characterizations of Applicative Structures, *Math. Struct. in Comp. Science* **9**, 1999.

[HLMP98] F.Honsell, M.Lenisa, U.Montanari, M.Pistore. Final Semantics for the π-calculus, *PROCOMET'98* Conf. Proc., D. Gries et al. eds, Chapman & Hall, 1998.

[How96] D.Howe. Proving congruence of bisimulation in functional programming languages, *Inf. and Comp.* **124**(2), 1996, 103–112.

[Jac96] B.Jacobs. Objects and Classes, co-algebraically, *Object-Orientation with Parallelism and Book Persistence*, B.Freitag et al. eds., Kluwer Academic Publishers, 1996.

[Jac97] B.Jacobs. Behaviour-refinement of object-oriented specifications with coinductive correctness proofs, *TAPSOFT'97*, M.Bidoit, et. al. eds., Springer LNCS **1214**, 1997, 787–802.

[Las96] S.Lassen. Relational Reasoning About Contexts, in *Higher Order Operational Techniques in Semantics*, A.Gordon and A.Pitts eds., Cambridge University Press.

[Las99] S.Lassen. Bisimulation in Untyped Lambda Calculus: Böhm Trees and Bisimulation up to Context, *MFPS XV* Conf. Proc., ENTCS, 1999.

[Len96] M.Lenisa. Final Semantics for a Higher Order Concurrent Language, *CAAP'96*, H.Kirchner et. al. eds., Springer LNCS **1059**, 1996, 102–118.

[Len97] M.Lenisa. Semantic Techniques for Deriving Coinductive Characterizations of Observational Equivalences for λ-calculi, *TLCA'97* Conference Proceedings, P.de Groote, R.Hindley, eds., Springer LNCS **1210**, 1997, 63–81.

[Len97a] M.Lenisa. A Uniform Syntactical Method for Proving Coinduction Principles in λ-calculi, *TAPSOFT'97*, M.Bidoit, et. al. eds., Springer LNCS **1214**, Berlin 1997, 248–266.

[Len98] M.Lenisa. Themes in Final Semantics, Ph.D. Thesis **TD-6/98**, Dipartimento di Informatica, Università di Pisa, March 1998.

[Len99] M.Lenisa. A Complete Coinductive Logical System for Bisimulation Equivalence on Circular Objects, to appear in *FoSSaCS'99* (*ETAPS*) Conf. Proc., W.Thomas ed., Springer LNCS **1578**, 1999, 243–257.

[MST96] I.Mason, S.Smith, C.Talcott. From Operational Semantics to Domain Theory, *Inf. and Comp.* **128**(1), 1996, 26–47.

[Mil83] R.Milner. Calculi for synchrony and asynchrony, *TCS* **25**, 1983, 267–310.

[MPW92] R.Milner, J.Parrow, D.Walker. A calculus of mobile processes, *Inf. and Comp.* **100**, 1992, 1–77.

[Mos] L.Moss. Coalgebraic Logic, to appear in the *Annals of Pure and Applied Logic.*

[Par81] D.M.R.Park. Concurrency and automata on infinite sequences, *MFPS'93*, P.Deussen ed., Springer LNCS **104**, 1981, 167–183.

[Pit96] A.M.Pitts. Relational Properties of Domains, *Inf. and Comp.* **127**, 1996, 66–90.

[Pit96a] A.M.Pitts. A note on logical relations between semantics and syntax, *J. of the Interest Group in Pure and Applied Logics*, Special Issue for the 3rd *WoLLIC'96* Workshop, 1996.

[Plo85] G.Plotkin. Types and Partial Functions, Post-Graduate Lecture Notes, Department of Computer Science, University of Edinburgh, 1985.

[PT99] J.Power, D.Turi. A Coalgebraic Foundation for Linear Time Semantics, *CTCS'99* Conf. Proc., M.Hofmann et al. eds., 1999, 338–354.

[Rut96] J.J.M.M.Rutten. Universal coalgebra: a theory of systems, **CS-R9652**, CWI, 1996.

[Rut98] J.J.M.M.Rutten. Relators and Metric Bisimulations, *CMCS'98* Conf. Proc., ENTCS **11**, 1998.

[RT93] J.J.M.M.Rutten, D.Turi. On the Foundations of Final Semantics: Non-Standard Sets, Metric Spaces, Partial Orders, *REX* Conference Proceedings, J.de Bakker et al. eds., Springer LNCS **666**, 1993, 477–530.

[RT94] J.J.M.M.Rutten, D.Turi. Initial algebra and final coalgebra semantics for concurrency: A Decade of Concurrency - Reflections and Perspectives, *REX* Conference Proceedings, J.de Bakker et al. eds., Springer LNCS **803**, 1994, 530–582.

[RV97] J.J.M.M.Rutten, E.de Vink, Bisimulation for probabilistic transition systems: a coalgebraic approach (extended abstract), *ICALP'97* Conference Proceedings, P.Degano et al, eds., Springer LNCS **1256**, 1997, 460–470.

[San98] D.Sangiorgi. On the bisimulation proof method, *Math. Struct. In Comp. Science* **98**(8), 1998, 447–478.

[SP82] M.Smyth, G.Plotkin. The category-theoretic solution of recursive domain equations, *SIAM J. of Computing* **11**(4), 1982, 761–783.

[Tur96] D.Turi. *Functorial Operational Semantics and its Denotational Dual*, PhD thesis, 1996.

[TP97] D.Turi, G.Plotkin. Towards a mathematical operational semantics, Proc 12^{th} *LICS* Conf., IEEE, Computer Science Press, 1997, 280–291.