Corecursive Algebras: A Study of General Structured Corecursion

Venanzio Capretta[1], Tarmo Uustalu[2], and Varmo Vene[3]

[1] School of Computer Science, University of Nottingham, United Kingdom
[2] Institute of Cybernetics at Tallinn University of Technology, Estonia
[3] Department of Computer Science, University of Tartu, Estonia

Abstract. Motivated by issues in designing practical total functional programming languages, we are interested in structured recursive equations that uniquely describe a function not because of the properties of the coalgebra marshalling the recursive call arguments but thanks to the algebra assembling their results. Dualizing the known notions of recursive and wellfounded coalgebras, we call an algebra of a functor corecursive, if from any coalgebra of the same functor there is a unique map to this algebra, and antifounded, if it admits a bisimilarity principle. Differently from recursiveness and wellfoundedness, which are equivalent conditions under mild assumptions, corecursiveness and antifoundedness turn out to be generally inequivalent.

1 Introduction

In languages of total functional programming [17], such as Cockett's Charity [5] and type-theoretic proof assistants and dependently typed languages, unrestricted general recursion is unavailable. Instead, these languages support structured recursion and corecursion schemes for defining functions with inductive domains resp. coinductive codomains. For inductive types such schemes include iteration, primitive recursion and recursion on structurally smaller arguments ("guarded-by-destructors" recursion). Programming with coinductive types can dually be supported, e.g., by "guarded-by-constructors" corecursion [6,10].

Characteristically, schemes like this define a function as the unique solution of a recursive equation where the right-hand side marshals the arguments of recursive calls, makes the recursive calls and assembles their results. Operational intuition tells us that structured recursion defines a function uniquely as the argument is gradually consumed and structured corecursion because the result is gradually produced. More generally, instead of structurally smaller recursive call arguments one can allow arguments smaller in the sense of some wellfounded relation (not necessarily on an inductive type). We may ask: Does the "productivity" aspect of structured corecursion admit a similar generalization? What are some principles for reasoning about functions defined in this way? In this article we address exactly these questions in an abstract categorical setting.

General structured recursion and induction have been analysed in terms of recursive and wellfounded coalgebras. A *recursive coalgebra* (RCA) is a coalgebra

M.V.M. Oliveira and J. Woodcock (Eds.): SBMF 2009, LNCS 5902, pp. 84–100, 2009.

of an endofunctor F with a unique coalgebra-to-algebra morphism to any F-algebra. In other words, it is a coalgebra that guarantees unique solvability of any structured recursive diagram involving it. This abstract version of the wellfounded recursion principle was introduced by Osius [13]. It was also of interest to Eppendahl [9], and we have previously studied constructions to obtain recursive coalgebras from other coalgebras already known to be recursive, with the help of distributive laws of functors over comonads [4].

Taylor introduced the notion of *wellfounded coalgebra* (WFCA), an abstract version of the wellfounded induction principle, and proved that, under mild assumptions, it is equivalent to RCA [14,15],[16, Ch. 6]. Defined in terms of Jacobs's next-time operator [11], a wellfounded coalgebra is a coalgebra such that any subset of its carrier containing its next-time subset is isomorphic to the carrier, so that the carrier is the least fixed-point of the next-time operator. As this least fixed-point is given by those elements of the carrier whose recursive calls tree is wellfounded, the principle really states that all of the carrier is included in the "wellfounded core" (cf. Bove-Capretta's method [2] in type theory: a general-recursive definition is made acceptable by casting it as a definition by structured recursion on the inductively defined wellfounded core and proving that the whole domain is in the wellfounded core). A closely related condition has the coalgebra carrier reconstructed by iterating the next-time operator on the empty set.

Adámek et al. [1] provided additional characterizations for the important case when the functor has an initial algebra. Backhouse and Doornbos [8] studied wellfoundedness in a relational setting.

We look at the dual notions with the aim to achieve a comparable analysis of structured corecursion and coinduction. It is only to be expected that several differences arise from the fact that Set-like categories are not self-dual. More surprisingly, however, they turn out to be quite deep. The dual of RCA is the notion of *corecursive algebra* (CRA): we call an algebra corecursive if there is a unique map to it from any coalgebra. Here the first discrepancy arises: while it is well-known that initial algebras support primitive recursion and, more generally, a recursive coalgebra is parametrically recursive ([16, Ch. 6]), the dual statement is not true: corecursiveness with the option of an escape (complete iterativity in the sense of Milius [12]) is a strictly stronger condition than plain corecursiveness.

The dual of WFCA is the notion of *antifounded algebra* (AFA)[1]. The dual of the next-time operator maps a quotient of the carrier of an algebra to the quotient identifying the results of applying the algebra structure to elements that were identified in the original quotient. AFA is a categorical formulation of the principle of bisimilarity: if a quotient is finer than its next-time quotient, then it must be isomorphic to the algebra carrier. Here also the equivalence with CRA is not satisfied: both implications fail for rather simple algebras in Set.

Finally, we call an algebra *focusing* (FA), if the algebra carrier can be reconstructed by iterating the dual next-time operator. In the coalgebra case, one

[1] Our choice of the name was motivated by the relation to the set-theoretic antifoundation axioms.

starts with the empty set and constructs a chain of ever larger subsets of the carrier. Now, we start with the singleton set, which is the quotient of the carrier by the total relation, and construct an inverse chain of ever finer quotients. Intuitively, each iteration of the dual next-time operator refines the quotient. And while a solution of a recursive diagram in the recursive case is obtained by extending the approximations to larger subsets of the intended domain, now it is obtained by sharpening the approximations to range over finer quotients of the intended codomain. FA happens to be the strongest of the conditions, implying both AFA and CRA. The inverse implications turn out to be false.

The article is organized around these three notions, treated in Sections 2, 3 and 4, respectively, before we arrive at our conclusions in Section 5. Throughout the article we are interested in conditions on an algebra (A, α) of an endofunctor F on a category $\mathcal{C}$. We assume that $\mathcal{C}$ has pushouts along epis and that F preserves epis.[2] Our prime examples are $\mathcal{C}$ being Set and F a polynomial functor.

2 Corecursive Algebras

Our central object of study in this article is the notion of corecursive algebra, the dual of Osius's concept recursive coalgebra [13].

Definition 1. *An algebra (A, α) of an endofunctor F on a category $\mathcal{C}$ is called* corecursive *(CRA) if for every coalgebra (C, γ) there exists a unique coalgebra-to-algebra map, i.e., a map $f : C \to A$ making the following diagram commute:*

$$\begin{array}{ccc} C & \xrightarrow{\gamma} & FC \\ {\scriptstyle f}\downarrow & & \downarrow{\scriptstyle Ff} \\ A & \xleftarrow[\alpha]{} & FA. \end{array}$$

We write separately CRA-existence and CRA-uniqueness for the statements that the diagram has at least and at most one solution, respectively.

An algebra is corecursive if every structured recursive diagram (= coalgebra-to-algebra map diagram) based on it defines a function (in the sense of turning out to be a definite description). The inverse of the final F-coalgebra, whenever it exists, is trivially a corecursive F-algebra (in fact the initial corecursive F-algebra). However, there are numerous examples of corecursive algebras that arise in different ways.

Example 1. We illustrate the definition with a corecursive algebra in Set, for the functor $FX = E \times X \times X$, where E is some fixed set. The carrier is the set of streams over E, $A = \mathsf{Str}(E)$. The algebra structure α is defined as follows:

$$\begin{array}{ll} \alpha : E \times \mathsf{Str}(E) \times \mathsf{Str}(E) \to \mathsf{Str}(E) & \qquad \mathsf{merge} : \mathsf{Str}(E) \times \mathsf{Str}(E) \to \mathsf{Str}(E) \\ \alpha(e, s_1, s_2) = e :: \mathsf{merge}(s_1, s_2) & \qquad \mathsf{merge}(e :: s_1, s_2) = e :: \mathsf{merge}(s_2, s_1). \end{array}$$

[2] In the recursive case it makes sense to additionally require that F preserves pullbacks along monos. This assumption holds for typical functors of interest. In the presence of a subobject classifier in $\mathcal{C}$, it guarantees that recursiveness of a coalgebra implies wellfoundedness. The dual assumption, that F preserves pushouts along epis, is not as helpful. Moreover, it is too strong: it is false, e.g., for $FX = X \times X$. We drop it.

It is easy to see that this algebra is corecursive, although it is not the inverse of the final F-coalgebra, which is the set of non-wellfounded binary trees with nodes labelled by elements of E.

A simple example of recursive definition that could be justified by the corecursiveness of (A, α) is the following. Let $E = 2^*$ (lists of bits, i.e., binary words). We define a F-coalgebra $(C, \gamma : C \to 2^* \times C \times C)$ by $C = 2^*$ and $\gamma(l) = (l, 0l, 1l)$. We can now be certain that there is exactly one function $f : 2^* \to \mathsf{Str}(2^*)$ such that $f = \alpha \circ Ff \circ \gamma$. This function sends a binary word to the lexicographical enumeration of the binary words which have this given one as a prefix. In particular, the stream $f(\varepsilon)$ is the lexicographical enumeration of all binary words.

Example 2. We also obtain a corecursive algebra by endowing $A = \mathsf{Str}(E)$ with the following algebra structure of the functor $FX = E \times X$ (note that this is different from the inverse of the final F-coalgebra structure also carried by A):

$$\begin{array}{ll} \alpha : E \times \mathsf{Str}(E) \to \mathsf{Str}(E) & \mathsf{double} : \mathsf{Str}(E) \to \mathsf{Str}(E) \\ \alpha(e, s) = e :: \mathsf{double}(s) & \mathsf{double}(e :: s) = e :: e :: \mathsf{double}(s). \end{array}$$

The next notion is an important variation.

Definition 2. *An algebra (A, α) is called* parametrically corecursive *(pCRA) if for every object C and map $\gamma : C \to FC + A$ (that is, coalgebra of $F(-) + A$), there exists a unique map $f : C \to A$ making the following diagram commute:*

$$\begin{array}{ccc} C & \xrightarrow{\gamma} & FC + A \\ {\scriptstyle f}\downarrow & & \downarrow{\scriptstyle Ff + \mathsf{id}_A} \\ A & \xleftarrow[{[\alpha, \mathsf{id}_A]}]{} & FA + A. \end{array}$$

This notion is known under the name of *completely iterative algebra* [12].[3] While this term is well-established and we do not wish to question its appropriateness in any way, we use a different term here, locally, for better fit with the topic of this article (the adjective "parametrically" remains idiosyncratic however).

To be parametrically corecursive, an algebra must define a function also from diagrams where, for some arguments, the value of the function is given by an "escape". The inverse of the final coalgebra always has this property [18]. Examples 1, 2 also satisfy pCRA. We leave the verification to the reader.

Proposition 1. *pCRA $\Rightarrow$ CRA : A parametrically corecursive coalgebra is corecursive.*

Proof. Given a coalgebra (C, γ), the unique solution of the pCRA diagram for the map $(C, C \xrightarrow{\gamma} FC \xrightarrow{\mathsf{inl}} FC + A)$ is trivially also the unique solution of the CRA diagram for (C, γ). $\square$

[3] In this terminology inspired by iterative theories, the word "iterative" refers to iteration in the sense of tail-recursion. "Completely iterative" means that a unique solution exists for every coalgebra while "iterative" refers to the existence of such solutions only for finitary coalgebras, i.e., coalgebras with finitary carriers.

The following counterexamples show that the converse is not true (differently from the dual situation of recursive and parametrically recursive coalgebras). We exhibit an algebra that is corecursive but not parametrically corecursive.

Example 3. In the category Set, we use the functor $FX = X \times X$. An interesting observation is that any corecursive algebra (A, α) for this F must have exactly one fixed point, that is, one element a such that $\alpha(a, a) = a$. We take the following algebra structure on the three-element set $A = 3 = \{0, 1, 2\}$:

$$\begin{array}{ll} \alpha : 3 \times 3 \to 3 & \\ \alpha(1, 2) \;\; = 2 & \\ \alpha(n, m) = 0 & \text{otherwise.} \end{array}$$

Proposition 2. *CRA $\not\Rightarrow$ pCRA-uniqueness: Example 3 is corecursive, but does not satisfy the uniqueness property for parametrically corecursive algebras.*

Proof. Example 3 satisfies CRA. Let (C, γ) be a coalgebra. We prove that the only possible solution f of the CRA diagram is the constant 0. In fact, for $c \in C$, it cannot be $f(c) = 1$, because 1 is not in the range of α. On the other hand, if $f(c) = 2$, then we must have $f(c) = \alpha((f \times f)(\gamma(c)))$. Let us call c_0 and c_1 the two components of $\gamma(c)$: $\gamma(c) = (c_0, c_1)$. Then we have $f(c) = \alpha(f(c_0), f(c_1))$. For $f(c)$ to be equal to 2, it is necessary that $f(c_0) = 1$ and $f(c_1) = 2$. But we already determined that $f(c_0) = 1$ is impossible. In conclusion, there is a unique solution: $f(c) = 0$ for every $c \in C$.

Example 3 does not satisfy pCRA-uniqueness. The pCRA diagram for $C = \mathbb{B}$ and $\gamma : \mathbb{B} \to \mathbb{B} \times \mathbb{B} + 3$ defined by $\gamma(\mathsf{true}) = \mathsf{inr}(1), \gamma(\mathsf{false}) = \mathsf{inl}(\mathsf{true}, \mathsf{false})$, has two distinct solutions:

$$\begin{array}{ccc} \mathbb{B} & \xrightarrow{\;\gamma\;} & \mathbb{B} \times \mathbb{B} + 3 \\ {\scriptstyle f_0} \downarrow \downarrow {\scriptstyle f_1} & & {\scriptstyle f_0 \times f_0 + \mathsf{id}} \downarrow \downarrow {\scriptstyle f_1 \times f_1 + \mathsf{id}} \\ 3 & \xleftarrow[{[\alpha, \mathsf{id}]}]{} & 3 \times 3 + 3 \end{array} \qquad \begin{array}{ll} f_0(\mathsf{true}) \;\, = 1 \\ f_0(\mathsf{false}) = 0 \end{array} \qquad \begin{array}{ll} f_1(\mathsf{true}) \;\, = 1 \\ f_1(\mathsf{false}) = 2. \end{array}$$

(Note that Example 3 satisfies pCRA-existence: to construct a solution, put it equal to 0 on all argument values on which it is not recursively forced.) □

Example 4. Consider the following algebra (A, α) for the functor $FX = X \times X$ in Set: We take A to be $\mathbb{N}$ and define the algebra structure by

$$\begin{array}{ll} \alpha : \mathbb{N} \times \mathbb{N} \to \mathbb{N} & \\ \alpha(1, m) = m + 2 & \\ \alpha(n, m) = 0 & \text{if } n \neq 1. \end{array}$$

Proposition 3. *CRA $\not\Rightarrow$ pCRA-existence: Example 4 is corecursive, but does not satisfy the existence property for parametrically corecursive algebras.*

Proof. Example 4 satisfies CRA, essentially by the same argument as for Example 3: the unique solution is forced to be the constant 0.

Example 4 does not satisfy pCRA-existence. Take $C = \mathbb{B}$ and define $\gamma : \mathbb{B} \to \mathbb{B} \times \mathbb{B} + \mathbb{N}$ by $\gamma(\mathsf{true}) = \mathsf{inr}(1)$ and $\gamma(\mathsf{false}) = \mathsf{inl}(\mathsf{true}, \mathsf{false})$. For this case, there is no solution to the pCRA diagram. Indeed, a solution should surely satisfy $f(\mathsf{true}) = 1$. Therefore we should also have $f(\mathsf{false}) = \alpha(f(\mathsf{true}), f(\mathsf{false})) = \alpha(1, f(\mathsf{false})) = f(\mathsf{false}) + 2$, which is clearly impossible.

(Note that Example 4 satisfies pCRA-uniqueness: the value of a solution $f(c)$ can be undetermined only if $\gamma(c) = \mathsf{inl}(c_1, c_2)$ with $f(c_1) = 1$ and $f(c_2)$ undetermined in turn. But this cannot go on forever because it would give an unbounded value.) □

3 Antifounded Algebras

Now we turn to the dual of Taylor's wellfounded coalgebras. We state the definition with the help of the dual of the next-time operator of Jacobs [11]. Remember that we assume that the category $\mathcal{C}$ has pushouts along epis and that F preserves epis.

Definition 3. *Given an algebra* (A, α)*. Let* $(Q, q : A \twoheadrightarrow Q)$ *be a quotient of* A *(i.e., an epi with* A *as the domain*[4]*). We define a new quotient* $(\mathsf{nt}_A(Q), \mathsf{nt}_A(q) : A \twoheadrightarrow \mathsf{nt}_A(Q))$ *(the* next-time *quotient) by the following pushout diagram:*

$$
\begin{array}{ccc}
A & \xleftarrow{\quad \alpha \quad} & FA \\
{\scriptstyle \mathsf{nt}_A(q)} \downarrow & & \downarrow {\scriptstyle Fq} \\
\mathsf{nt}_A(Q) & \xleftarrow[\quad \alpha[q] \quad]{} & FQ
\end{array}
$$

Note that $\mathsf{nt}_A(q)$ is guaranteed to be an epi, as a pushout along an epi.

Notice that we abuse notation (although in a fairly standard fashion): First, nt_A is really parameterized not by the object A, but the algebra (A, α). And further, nt_A operates on a quotient (Q, q) and returns another quotient given by the vertex and one of the side morphisms of the pushout. It is a convention of convenience to denote the vertex by $\mathsf{nt}_A(Q)$ and the morphism by $\mathsf{nt}_A(q)$.

In particular, in the category Set we can give an intuitive definition of nt_A in terms of quotients by equivalence relations. In Set, a quotient is, up to isomorphism, an epi $q : A \twoheadrightarrow A/{\equiv}$, where $\equiv$ is an equivalence relation on A, with $q(a) = [a]_{\equiv}$. Its next-time quotient can be represented similarly: $\mathsf{nt}_A(A/{\equiv}) = A/{\equiv'}$, where $\equiv'$ is the reflexive-transitive closure of the relation

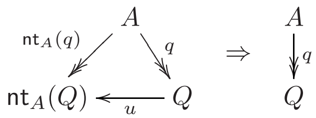
$$\{(\alpha(y_0), \alpha(y_1)) \mid y_0, y_1 \in FA, y_0\ (F{\equiv})\ y_1\}.$$

Here $F{\equiv}$ is the lifting of $\equiv$ to FA: it identifies elements of FA that have the same *shape* and equivalent elements of A in corresponding *positions* (if $\equiv$ is given by a span $(R, r_0, r_1 : R \to A)$, $F{\equiv}$ is just (FR, Fr_0, Fr_1)).

The following definition is the dual of Taylor's definition of *wellfounded algebra* [14,15,16].

[4] We do not bother to identify equivalent epis, see below.

Definition 4. *An algebra* (A, α) *is called* antifounded *(AFA) if for every quotient* $(Q, q : A \twoheadrightarrow Q)$*, if* $\mathsf{nt}_A(q)$ *factors through* q*, then* q *is an isomorphism. In diagrams:*



is an isomorphism.

Note that, if $\mathsf{nt}_A(q)$ factors, i.e., u exists, then it is necessarily unique, as q is an epi. Note also that q being an isomorphism means that id_A factorizes through q, i.e., that q is a split mono.

Example 1 is an antifounded algebra. Indeed, let $q : \mathsf{Str}(E) \twoheadrightarrow \mathsf{Str}(E)/{\equiv}$ be a quotient of $\mathsf{Str}(E)$ such that $\mathsf{nt}_A(q)$ factors through q. Let $\equiv'$ be the equivalence relation giving the next-time quotient, that is, $\mathsf{nt}_A(\mathsf{Str}(E)/{\equiv}) = \mathsf{Str}(E)/{\equiv'}$. It is the reflexive-transitive closure of the relation

$$\begin{aligned}\{(e :: \mathsf{merge}(s_{00}, s_{01}), e :: \mathsf{merge}(s_{10}, s_{11}))\\ \mid e \in E, s_{00}, s_{01}, s_{10}, s_{11} \in \mathsf{Str}(E), s_{00} \equiv s_{10} \wedge s_{01} \equiv s_{11}\}\end{aligned}$$

This relation is already reflexive and transitive, so the closure is in fact unnecessary. The hypothesis that $\mathsf{nt}_A(q)$ factors through q tells us that $\equiv$ is finer than $\equiv'$, that is, $\forall s_0, s_1 \in \mathsf{Str}(E).\, s_0 \equiv s_1 \Rightarrow s_0 \equiv' s_1$. We want to prove that $\equiv$ must be equality. In fact, suppose $s_0 \equiv s_1$, then also $s_0 \equiv' s_1$. This means that they must have the same head element e_0 and that their *unmerged* parts must be equivalent: if $s_{00}, s_{01}, s_{10}, s_{11}$ are such that $s_0 = e_0 :: \mathsf{merge}(s_{00}, s_{01})$ and $s_1 = e_0 :: \mathsf{merge}(s_{10}, s_{11})$, then it must be $s_{00} \equiv s_{10}$ and $s_{01} \equiv s_{11}$; repeating the argument for these two equivalences, we can deduce that s_0 and s_1 have the same second and third element, and so on. In conclusion, $s_0 = s_1$ as desired.

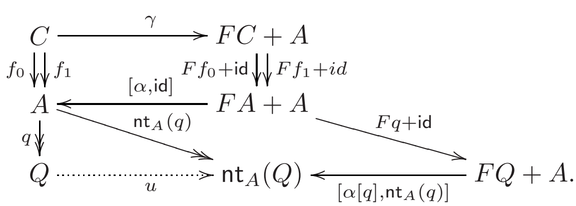
Example 2 can be seen to be an antifounded algebra by a similar argument. The next-time equivalence relation $\equiv'$ of an equivalence relation $\equiv$ on $\mathsf{Str}(E)$ is the reflexive closure of the transitive relation

$$\{(e :: \mathsf{double}(s_0), e :: \mathsf{double}(s_1)) \mid e \in E, s_0, s_1 \in \mathsf{Str}(E), s_0 \equiv s_1\}.$$

Theorem 1. *AFA* $\Rightarrow$ *pCRA-uniqueness: An antifounded algebra* (A, α) *satisfies the uniqueness part of the parametric corecursiveness condition.*

Proof. Assume that (A, α) satisfies AFA and let f_0 and f_1 be two solutions of the pCRA diagram for some $(C, \gamma : C \to FC + A)$. We must prove that $f_0 = f_1$.

Let $(Q, q : A \to Q)$ be the coequalizer of f_0 and f_1. As any coequalizer, it is epi. We apply the next-time operator to it. We prove that $\mathsf{nt}_A(q) \circ f_0 = \mathsf{nt}_A(q) \circ f_1$; the proof is summarized by this diagram:



By the fact that f_0 and f_1 are solutions of the pCRA diagram, the top rectangle commutes for both of them. By definition of the nt_A operator, the lower-right parallelogram commutes. Therefore, we have that $\mathsf{nt}_A(q) \circ f_0 = [\alpha[q] \circ F(q \circ f_0), \mathsf{nt}_A(q)] \circ \gamma$ and $\mathsf{nt}_A(q) \circ f_1 = [\alpha[q] \circ F(q \circ f_1), \mathsf{nt}_A(q)] \circ \gamma$. But $q \circ f_0 = q \circ f_1$, because q is the coequalizer of f_0 and f_1, so the right-hand sides of the two previous equalities are the same. We conclude that $\mathsf{nt}_A(q) \circ f_0 = \mathsf{nt}_A(q) \circ f_1$.

Now, using once more that q is the coequalizer of f_0, f_1, there must exist a unique map $u : Q \to \mathsf{nt}_A(Q)$ such that $u \circ q = \mathsf{nt}_A(q)$. By AFA, this implies that q is an isomorphism. As $q \circ f_0 = q \circ f_1$, it follows that $f_0 = f_1$. □

However, AFA does not imply CRA-existence (and therefore, does not imply pCRA-existence), as attested by the following counterexample.

Example 5. In Set, we use the identity functor $FX = X$ and the successor algebra on natural numbers: $A = \mathbb{N}$ and $\alpha : \mathbb{N} \to \mathbb{N}$ is defined by $\alpha(n) = n + 1$.

Proposition 4. *AFA $\not\Rightarrow$ CRA-existence: Example 5 satisfies AFA but not CRA-existence.*

Proof. Example 5 satisfies AFA. Let $q : A \to A/{\equiv}$ be a quotient of A such that $\mathsf{nt}_A(q)$ factorizes through q. Note that the definition of $\equiv'$ (the next-time equivalence relation of $\equiv$) is particularly simple, just the reflexive closure of

$$\{(m_0 + 1, m_1 + 1) \mid m_0, m_1 \in \mathbb{N}, m_0 \equiv m_1\}.$$

So two distinct numbers are equivalent according to $\equiv'$ if and only if they are the successors of elements that are equal according to $\equiv$. There is no need of a transitive closure in this case, since the relation is already transitive. By assumption $\equiv$ is finer than $\equiv'$, that is $\forall m_1, m_2 \in \mathbb{N}.\, m_0 \equiv m_1 \Rightarrow m_0 \equiv' m_1$. We want to prove that $\equiv$ is equality. We prove, by induction on m, that $[m]_\equiv = \{m\}$, that is, every equivalence class is a singleton:

- For $m = 0$ the statement is trivial: $0 \equiv m'$ implies, by hypothesis, that $0 \equiv' m'$, but since 0 is not a successor, this can happen only by reflexivity, that is, if $m' = 0$;
- Assume that $[m]_\equiv = \{m\}$ by induction hypothesis; we must prove that $[m+1]_\equiv = \{m+1\}$; if $m + 1 \equiv m'$, then $m + 1 \equiv' m'$, which can happen only if either $m + 1 = m'$ or m' is a successor and $m \equiv m' - 1$; by induction hypothesis, this implies that $m' - 1 = m$, so $m' = m + 1$.

Example 5 does not satisfy CRA-existence. Indeed, if we take the trivial coalgebra $(1 = \{0\}, \mathsf{id} : 1 \to 1)$, we see that a solution of the CRA diagram would require $f(0) = f(0) + 1$, which is impossible. □

The vice versa also does not hold: CRA does not imply AFA, as evidenced by the following counterexample.

Example 6. We use the functor $FX = 2^* \times X$ in Set, where 2^* is the set of lists of bits (binary words). We construct an F-algebra on the carrier $A =$

$\mathsf{Str}(2^*)/\simeq$, where $\simeq$ is the equivalence relation defined below. We are particularly interested in streams of a special kind: those made of *incremental* components that stabilize after at most one step. Formally, if $l \in 2^*$ and $i, j \in 2$, we define $(l)^{i\bar{j}} = (li, lij, lijj, lijjj, \ldots)$, that is,

$$(l)^{0\bar{0}} = (l0, l00, l000, l0000, l00000, \ldots) \quad (l)^{0\bar{1}} = (l0, l01, l011, l0111, l01111, \ldots)$$
$$(l)^{1\bar{0}} = (l1, l10, l100, l1000, l10000, \ldots) \quad (l)^{1\bar{1}} = (l1, l11, l111, l1111, l11111, \ldots).$$

The relation $\simeq$ is the least congruence such that $(l)^{0\bar{1}} \simeq (l)^{1\bar{0}}$ for every l. This means that two streams that begin in the same way but then stabilize in one of the two forms above will be equal: $(l_0, \ldots, l_{k-1}) +\!\!+ (l)^{0\bar{1}} \simeq (l_0, \ldots, l_{k-1}) +\!\!+ (l)^{1\bar{0}}$. In other words, the equivalence classes of $\simeq$ are $\{(l_0, \ldots, l_{k-1}) +\!\!+ (l)^{0\bar{1}}, (l_0, \ldots, l_{k-1}) +\!\!+ (l)^{1\bar{0}}\}$ for elements in one of those two forms, and singletons for elements not in those forms. Notice that we do not equate elements of the forms $(l)^{0\bar{0}}$ and $(l)^{1\bar{1}}$. For simplicity, we will write elements of A just as sequences, in place of equivalence classes. So if $s \in \mathsf{Str}(2^*)$, we will use s also to indicate $[s]_\simeq$. We leave it to the reader to check that all our definitions are invariant with respect to $\simeq$. We now define an algebra structure α on this carrier by:

$$\alpha : 2^* \times (\mathsf{Str}(2^*)/\simeq) \to \mathsf{Str}(2^*)/\simeq$$
$$\alpha(l, s) = l :: s.$$

Proposition 5. *pCRA $\not\Rightarrow$ AFA: Example 6 satisfies pCRA but not AFA.*

Proof. First we prove that Example 6 satisfies pCRA. Given some $(C, \gamma : C \to 2^* \times C + A)$, we want to prove that there is a unique solution to the pCRA diagram. Given any element $c : C$, we have two possibilities: $\gamma\, c = \mathsf{inr}\, s$, in which case it must necessarily be $f\, c = s$; or $\gamma\, c = \mathsf{inl}\langle l_0, c_1\rangle$, in which case it must be $f\, c = l_0 :: (f\, c_1)$. In this second case, we iterate γ again on c_1. The kth component of $f\, c$ is decided after at most k such steps, therefore the result is uniquely determined by commutativity of the diagram.

Now we prove that Example 6 does not satisfy AFA. With this goal we define an equivalence relation $\equiv$ on $A = \mathsf{Str}(2^*)/\simeq$ such that $\mathsf{nt}_A(A/\equiv)$ factorizes through $A/\equiv$ but $\equiv$ is strictly coarser than $\simeq$. The relation $\equiv$ is the reflexive closure of the following: $\forall l \in 2^*, i_0, i_1, j_0, j_1 \in 2.\, (l)^{i_0\bar{j_0}} \equiv (l)^{i_1\bar{j_1}}$. In other words, $\equiv$ identifies all elements in the form $(l)^{i\bar{j}}$ that have the same base sequence l. Contrary to the case of $\simeq$, we do not extend $\equiv$ to a congruence: $l_0 +\!\!+ (l_1)^{0\bar{0}} \not\equiv l_0 +\!\!+ (l_1)^{1\bar{1}}$, but still $l_0 +\!\!+ (l_1)^{0\bar{1}} \equiv l_0 +\!\!+ (l_1)^{1\bar{0}}$, because these elements are equivalent according to $\simeq$ and $\equiv$ is coarser. So if s_0 is not in the form $(l)^{i\bar{j}}$, then $s_0 \equiv s_1$ is true only if $s_0 \simeq s_1$. This equivalence relation is strictly coarser than $\simeq$, since $(l)^{0\bar{0}} \equiv (l)^{1\bar{1}}$ but $(l)^{0\bar{0}} \not\simeq (l)^{1\bar{1}}$.

Let $\equiv'$ be the next-time equivalence relation of $\equiv$, i.e., such that $\mathsf{nt}_A(A/\equiv) = A/\equiv'$. Concretely, $\equiv'$ is the (already reflexive and transitive) relation

$$\{(l :: s_0, l :: s_1) \mid l \in 2^*, s_0, s_1 \in A, s_0 \equiv s_1\}.$$

We prove that $\equiv$ is finer than $\equiv'$, i.e., if $s_0 \equiv s_1$, then $s_0 \equiv' s_1$. There are two cases.

If s_0 or s_1 is not in the form $(l)^{i\bar{j}}$, then its equivalence class is a singleton by definition, so the other element must be equal to it and the conclusion follows by reflexivity.

If both s_0 and s_1 are in the form $(l)^{i\bar{j}}$, then their base element must be the same l, by definition of $\equiv$. There are four cases for each of the two elements, according to what their i and j parameters are. By considerations of symmetry and reflexivity, we can reduce the cases to just two:

– $s_0 = (l)^{0\bar{0}}$ and $s_1 = (l)^{0\bar{1}}$: We can write the two elements alternatively as $s_0 = l0 :: (l0)^{0\bar{0}}$ and $s_1 = l0 :: (l0)^{1\bar{1}}$; since $(l0)^{0\bar{0}} \equiv (l0)^{1\bar{1}}$, we conclude that $s_0 \equiv' s_1$;
– $s_0 = (l)^{0\bar{0}}$ and $s_1 = (l)^{1\bar{1}}$: By the previous case and its dual we have $s_0 \equiv' (l)^{0\bar{1}}$ and $s_1 \equiv' (l)^{1\bar{0}}$; but $(l)^{0\bar{1}} \simeq (l)^{1\bar{0}}$ so $s_0 \equiv' s_1$ by transitivity. □

We now turn to a higher-level view of antifounded algebras. This is in terms of the classical fixed point theory for preorders and monotone endofunctions.

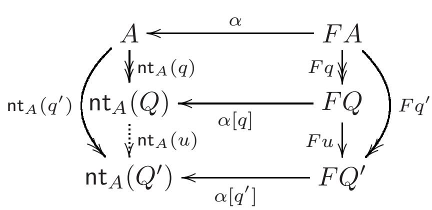
For a (locally small) category $\mathcal{C}$ and an object A, we define the category of quotients of A, called $\mathbf{Quo}(A)$ as follows:

– an object is an epimorphism $(Q, q : A \twoheadrightarrow Q)$,
– a map between (Q, q), (Q', q') is a map $u : Q \to Q'$ such that $u \circ q = q'$.

Clearly there can be at most one map between any two objects, so this category is a preordered set. (In the standard definition of the category, equivalent epis are identified, so it becomes a poset. We have chosen to be content with a preorder; the cost is that universal properties define objects up to isomorphism.) We tend to write $Q \leq Q'$ instead of $u : (Q, q) \to (Q', q')$, leaving q, q' and the unique u implicit.

Clearly, $\mathbf{Quo}(A)$ has (A, id_A) as the initial and $(1, !_A)$ as the final object.

Now, nt_A sends objects of $\mathbf{Quo}(A)$ to objects of $\mathbf{Quo}(A)$. It turns out that it can be extended to act also on maps. For a map $u : (Q, q) \to (Q', q')$, we define $\mathsf{nt}_A(u) : (\mathsf{nt}_A(Q), \mathsf{nt}_A(q)) \to (\mathsf{nt}_A(Q'), \mathsf{nt}_A(q'))$ as the unique map from a pushout, as shown in the following diagram:



Given that $\mathbf{Quo}(A)$ is a preorder, this makes nt_A trivially a functor (preservation of the identities and composition is trivial). In preorder-theoretic terms, we say that nt_A is a monotone function.

We can notice that (A, id_A) is trivially a fixed point of nt_A. Since it is the least element of $\mathbf{Quo}(A)$, it is the least fixed point.

The condition of (A, α) being antifounded literally says that, for any Q, $Q \leq \mathsf{nt}_A(Q)$ implies $Q \leq A$, i.e., that A is an upper bound on the post-fixed points of nt_A. Taking into account that A, by being the least element, is also trivially a post-fixed point, this amounts to A being the greatest post-fixed point. Fixed point theory (or, if you wish, Lambek's lemma) tells us that the greatest post-fixed point is also the greatest fixed point.

So, in fact, (A, α) being antifounded means that (A, id_A) is a unique fixed point of nt_A. (Recall that this is up to isomorphism.)

4 Focusing Algebras

Our third and last notion of focused algebra, introduced below in Def. 6, is the condition that an algebra is recoverable by iterating its next-time operator, starting with the final quotient.

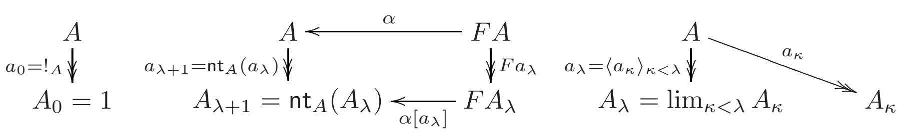
At transfinite iterations, given by limits in $\mathcal{C}$ (so that we can prove Theorem 3), we are not guaranteed to still obtain a quotient. In Prop. 7 we will prove that, for Example 5, the iteration at stage ω is not a quotient anymore. However, to apply fixed point theory to $\mathbf{Quo}(A)$ in Lemma 6, we need to work with limits in $\mathbf{Quo}(A)$. Below, talking about the iteration at a limit ordinal, we require that it is a quotient (assuming that so are also all preceding stages), or else we take it to be undefined. Clearly, this is not a beautiful definition. We regard it as one possible way to partially reconcile the discrepancy between corecursiveness and antifoundedness that we have already witnessed.

Definition 5. *Given an algebra (A, α), for any ordinal λ we partially define (A_λ, a_λ) (the λ-th iteration of nt_A on the final object $(1, !_A)$ of $\mathbf{Quo}(A)$) and maps $p_\lambda : A_{\lambda+1} \to A_\lambda$, $p_{\lambda,\kappa} : A_\lambda \to A_\kappa$ (for λ a limit ordinal and $\kappa < \lambda$) in $\mathcal{C}$ by simultaneous recursion by*

$$\begin{array}{lll}
A_0 = 1 & A_{\lambda+1} = \mathsf{nt}_A(A_\lambda) & A_\lambda = \lim_{\kappa<\lambda} A_\kappa \\
a_0 = !_A & a_{\lambda+1} = \mathsf{nt}_A(a_\lambda) & a_\lambda = \langle a_\kappa \rangle_{\kappa<\lambda} \\
p_0 = !_{A_1} & p_{\lambda+1} = \mathsf{nt}_A(p_\lambda) & p_\lambda = \langle p_\kappa \circ \mathsf{nt}_A(p_{\lambda,\kappa}) \rangle_{\kappa<\lambda} \\
 & & p_{\lambda,\kappa} = \pi_{\lambda,\kappa} \quad \text{if } \kappa < \lambda
\end{array}$$

where the third column applies if λ is a limit ordinal, the limit $\lim_{\kappa<\lambda} A_\kappa$ exists and the mediating map $\langle a_\kappa \rangle_{\kappa<\lambda}$ is epi; otherwise A_λ, a_λ, p_λ, and $p_{\lambda,\kappa}$ are left undefined.

Diagrammatically,



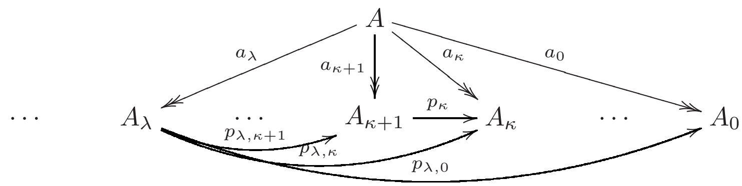
The limit in the limit ordinal case is of the following diagram in $\mathcal{C}$:

$$(A_\kappa, p_\kappa, p_{\kappa,\iota} \ (\kappa \text{ lim. ord.}, \iota < \kappa))_{\kappa<\lambda}.$$

Lemma 1. *The above definition is well-formed: for any* λ*,*

1. a_λ *is an epi (so, for any* λ*,* nt_A *is applicable to* (A_λ, a_λ)*, ensuring* $(A_{\lambda+1}, a_{\lambda+1})$ *is defined),*
2. $p_\lambda \circ a_{\lambda+1} = a_\lambda$ *and* $p_{\lambda,\kappa} \circ a_\lambda = a_\kappa$ *(if* λ *is a limit ordinal,* $\kappa < \lambda$*) (so, for any* λ*,* $(A, (a_\kappa)_{\kappa<\lambda})$ *in the definition of* a_λ *for* λ *a limit ordinal form a cone, ensuring* a_λ *is defined)*

Diagrammatically,



Proof. Both parts are proved by induction on λ.

(1) $a_0 = !_A$ is an epi. For the successor case, $a_{\lambda+1} = \mathsf{nt}_A(a_\lambda)$ is an epi, since nt_A takes quotients of A to quotients of A. Finally, for λ a limit ordinal, we have agreed to define a_λ as $\langle a_\kappa \rangle_{\kappa<\lambda}$ only if this mediating map is epi, leaving it undefined otherwise.

(2) It is trivial that $p_0 \circ a_1 = !_{A_1} \circ a_1 = !_A = a_0$.

For the successor case, $p_{\lambda+1} \circ a_{\lambda+2} = \mathsf{nt}_A(p_\lambda) \circ \mathsf{nt}_A(a_{\lambda+1}) = \mathsf{nt}_A(a_\lambda) = a_{\lambda+1}$ holds by the induction hypothesis $p_\lambda \circ a_{\lambda+1} = a_\lambda$, implying $\mathsf{nt}_A(p_\lambda) \circ \mathsf{nt}_A(a_{\lambda+1}) = \mathsf{nt}_A(a_\lambda)$ by the definition of the functorial extension of nt_A.

For λ a limit ordinal, $p_\lambda \circ a_{\lambda+1} = \langle p_\kappa \circ \mathsf{nt}_A(p_{\lambda,\kappa}) \rangle_{\kappa<\lambda} \circ \mathsf{nt}_A(a_\lambda) = \langle p_\kappa \circ \mathsf{nt}_A(p_{\lambda,\kappa}) \circ \mathsf{nt}_A(a_\kappa) \rangle_{\kappa<\lambda} = \langle p_\kappa \circ \mathsf{nt}_A(a_\kappa) \rangle_{\kappa<\lambda} = \langle p_\kappa \circ a_{\kappa+1} \rangle_{\kappa<\lambda} = \langle a_\kappa \rangle_{\kappa<\lambda} = a_\lambda$, from the induction hypotheses $p_{\lambda,\kappa} \circ a_\lambda = a_\kappa$, implying $\mathsf{nt}_A(p_{\lambda,\kappa}) \circ \mathsf{nt}_A(a_\lambda) = \mathsf{nt}_A(a_\kappa)$ by the definition of the functorial extension of nt_A, and from the induction hypotheses $p_\kappa \circ a_{\kappa+1} = a_\kappa$.

For λ a limit ordinal and $\kappa < \lambda$, $p_{\lambda,\kappa} \circ a_\lambda = \pi_{\lambda,\kappa} \circ \langle a_\kappa \rangle_{\kappa<\lambda} = a_\kappa$. □

It is very important to remember that we only accept $\lim_{\kappa<\lambda} A_\kappa$ (which is a limit in $\mathcal{C}$) as A_λ for λ a limit ordinal, if it is a quotient of A (otherwise we take A_λ to be undefined). This is by no means guaranteed. As the next proposition shows, this implies that A_λ is also a limit in $\mathbf{Quo}(A)$, but the vice versa need not be true. The carrier of a limit in $\mathbf{Quo}(A)$ is not necessarily a limit in $\mathcal{C}$, as evidenced by our analysis of Example 5 below.

Proposition 6. *If* A_λ *is defined for a limit ordinal (meaning that* $(A_\lambda, (p_{\lambda,\kappa})_{\kappa<\lambda})$ *is a limiting cone in* $\mathcal{C}$ *and* $a_\lambda = \langle a_\kappa \rangle_{\kappa<\lambda}$ *is epi), then* $((A_\lambda, a_\lambda), (p_{\lambda,\kappa})_{\kappa<\lambda})$ *is a limiting cone in* $\mathbf{Quo}(A)$*.*

Proof. To see that

$$((A_\kappa, a_\kappa), p_\kappa, p_{\kappa,\iota}\ (\kappa \text{ lim. ord.}, \iota < \kappa))_{\kappa<\lambda}$$

is a diagram in $\mathbf{Quo}(A)$ we need that $p_\kappa \circ a_{\kappa+1} = a_\iota$ and $p_{\kappa,\iota} \circ a_\kappa = a_\iota$ (κ a limit ordinal, $\iota < \kappa$) for $\kappa < \lambda$. To see that $((A_\lambda, a_\lambda), (p_{\lambda,\kappa})_{\kappa<\lambda})$ is a cone we also need $p_{\lambda,\kappa} \circ a_\lambda = a_\kappa$. But we have proved these equalities already.

To see that $((A_\lambda, a_\lambda), (p_{\lambda,\kappa})_{\kappa<\lambda})$ is a limiting cone, we observe that the sole map to it from a cone $((Q, q), (f_{\lambda,\kappa})_{\kappa<\lambda})$ in $\mathbf{Quo}(A)$ is given by the unique map from $(Q, (f_{\lambda,\kappa})_{\kappa<\lambda})$ to $(A_\lambda, (p_{\lambda,\kappa})_{\kappa<\lambda})$ in $\mathcal{C}$. □

Given that $\mathbf{Quo}(A)$ is a preorder, we have learned that $(A_\kappa)_{\kappa<\lambda}$ is an inverse chain (if all A_κ are defined) and the limit is the infimum.

Lemma 2. *If A_λ is defined and $A_\lambda \leq A_{\lambda+1}$, then A_λ is the greatest fixed point of nt_A.*

Proof. This is standard fixed point theory for preorders. A_λ is a post-fixed point of nt_A, as $A_\lambda \leq A_{\lambda+1} = \mathsf{nt}_A(A_\lambda)$. And by induction one checks that $Q \leq A_\kappa$ holds for any post-fixed point Q of nt_A and any κ: $Q \leq 1 = A_0$ is trivial; $Q \leq \mathsf{nt}_A(Q) \leq \mathsf{nt}_A(A_\kappa) = A_{\kappa+1}$ follows from the induction hypothesis $Q \leq A_\kappa$, as nt_A is monotone; and, finally, $Q \leq \inf_{\iota<\kappa} A_\iota$ is immediate from the induction hypotheses $Q \leq A_\iota$ $(\iota < \kappa)$. □

Definition 6. *(A, α) is* λ-focusing *(λ-FA) if A_λ is defined and $A_\lambda \cong A$.*

We show that Example 1 is ω-focusing. In fact we claim that, in this case, $A_i = \mathsf{Str}(E)/{\equiv_i}$, where $\equiv_i$ is the equivalence relation defined by $s_0 \equiv_i s_1$ if the first $2^i - 1$ elements of s_0 and s_1 are the same. The claim is clearly true for $i = 0$, because $\equiv_0$ is the total relation. Assume, as an induction hypothesis, that $A_i = \mathsf{Str}(E)/{\equiv_i}$. Then $A_{i+1} = \mathsf{nt}_A(\mathsf{Str}(E)/{\equiv_i}) = \mathsf{Str}(E)/{\equiv_{i+1}}$. Now $s_0 \equiv_{i+1} s_1$ holds if $s_0 = e_0 :: \mathsf{merge}(s_{00}, s_{01})$ and $s_1 = e_0 :: \mathsf{merge}(s_{10}, s_{11})$ with $s_{00} \equiv_i s_{10}$ and $s_{01} \equiv_i s_{11}$. By the induction hypothesis, this means that the first $2^i - 1$ elements of s_{00} and s_{10} are the same and the first $2^i - 1$ elements of s_{01} and s_{11} are also the same. In conclusion, the first $1 + (2^i - 1) + (2^i - 1) = 2^{i+1} - 1$ elements of s_0 and s_1 are the same, that is $s_0 \equiv_{i+1} s_1$, as claimed.

We have proved that A_i is isomorphic to the set E^{2^i-1} of vectors of length $2^i - 1$, with p_i the projection giving the first $2^i - 1$ elements of a vector of length $2^{i+1} - 1$. Standard reasoning shows that $\lim_{i<\omega} A_i$ is $\mathsf{Str}(E)$.

Example 2 is also ω-focusing, but the equivalence relations $\equiv_i$ are different. For $s_0 \equiv_i s_1$ to hold, if $s_0 \neq s_1$, it is not enough that they share the first $2^i - 1$ elements, say $e_0, \ldots, e_{2^i-2}$. It must moreover be the case that $e_1 = e_2$, $e_3 = e_4 = e_5 = e_6$, …, $e_{2^{i-1}-1} = \ldots = e_{2^i-2}$ and the remainders of s_0 and s_1 must both be in the image of double^i, i.e., consist of groups of 2^i equal elements.

There are examples of λ-focusing algebras that do not converge at the first limit ordinal ω but at later stages. Here is an example that converges at 2ω.

Example 7. Let us use the functor $FX = X + \mathbb{N} \times X$ in Set. We define an F-algebra with carrier $A = 2\omega + 1 = \{0, 1, \ldots, \omega, \omega + 1, \omega + 2, \ldots, 2\omega\}$:

$$\begin{aligned}
&\alpha : (2\omega + 1) + \mathbb{N} \times (2\omega + 1) \to 2\omega + 1 \\
&\alpha(\mathsf{inl}(x)) \quad\;\; = x + 1 \\
&\alpha(\mathsf{inr}(n, x)) = \min(\omega + x - n, 2\omega).
\end{aligned}$$

Theorem 2. *λ-FA $\Rightarrow$ AFA: If an algebra (A, α) is λ-focusing, it is antifounded.*

Proof. Assume that (A, α) is λ-focusing, i.e., that A_λ is defined and $A_\lambda \cong A$. Then $A_\lambda \cong A \leq A_{\lambda+1}$ trivially, as A is the least element in the preorder $\mathbf{Quo}(A)$. It follows by the previous lemma that A_λ, which is isomorphic to A, is the greatest fixed point of nt_A, i.e., that (A, α) is antifounded. □

The converse does not hold: Some antifounded algebras are not focusing.

Proposition 7. *AFA $\not\Rightarrow \exists\lambda.\, \lambda$-FA: Ex. 5 satisfies AFA but not λ-FA for any λ.*

Proof. We already proved in Proposition 4 that Example 5 satisfies AFA. Now we prove that it is not focusing at any ordinal. In fact, we have the following sequence of iterations of nt_A:

$$\begin{array}{l} A_0 = \{\bot\}, \quad A_1 = \{0, \bot\}, \quad A_2 = \{0, 1, \bot\}, \quad \ldots, \\ A_i = \{0, \ldots, i-1, \bot\}, \quad \ldots, \quad \lim_{i<\omega} A_i = \mathbb{N} \cup \{\bot\}. \end{array}$$

At the limit, the element $\bot$ is not an equivalence class of natural numbers anymore and the limit $\lim_{i<\omega} A_i$ is not a quotient of $A = \mathbb{N}$. So, in this case, the limit exists in Set, but is not a limit in the quotient category $\mathbf{Quo}(A)$. The reason that this happens is that, the limit in Set of an inverse chain of quotients given by equivalence relations is not necessarily the quotient given by the intersection of these equivalence relations. □

Notice that $(A_i)_{i<\omega}$ has the limit $\mathbb{N}$ in $\mathbf{Quo}(A)$. So we have to be mindful of the subtle distinction: λ-FA states that the limit exists in $\mathcal{C}$ and happens to be a quotient; this is a strictly stronger requirement than the condition that the limit exists in $\mathbf{Quo}(A)$.

Theorem 3. *λ-FA $\Rightarrow$ pCRA: If an algebra (A, α) is λ-focusing, it is parametrically recursive.*

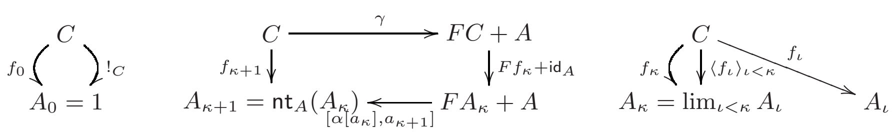
The proof uses the inverse chain $(A_\kappa)_{\kappa<\lambda+1}$ as the sequence of codomains for *fuzzy* approximations of the solution. The fact that $A = A_\lambda$ is the inverse limit establishes that a (sharp) function is achieved. This is analogous to the dual situation where a (total) solution arises from a sequence of *partial* approximations defined on a chain of subsets of the given domain to which the chain is required to have as the direct limit.

Proof. Assume that (A, α) is λ-focusing, i.e., that A_λ is defined and $A_\lambda = A_{\lambda+1} = A$ (we ignore that in general we have isomorphisms, not equalities).

Given $(C, \gamma : C \to FC + A)$, we define, for any κ, a map $f_\kappa : C \to A_\kappa$ by

$$\begin{array}{rll} f_0 &= !_C & \\ f_{\kappa+1} &= [\alpha[a_\kappa] \circ F f_\kappa, a_{\kappa+1}] \circ \gamma & \\ f_\kappa &= \langle f_\iota \rangle_{\iota<\kappa} & \text{if } \kappa \text{ is a lim. ord.} \end{array}$$

Diagrammatically,



Simultaneously, we show that $p_\kappa \circ f_{\kappa+1} = f_\kappa$ and $p_{\kappa,\iota} \circ f_\kappa = f_\iota$.

The base case $p_0 \circ f_1 = !_{A_1} \circ f_1 = !_C = f_0$ holds trivially.

For the successor case, we conclude $p_{\kappa+1} \circ f_{\kappa+2} = \mathsf{nt}_A(p_\kappa) \circ [\alpha[a_{\kappa+1}] \circ F f_{\kappa+1},$ $]a_{\kappa+2} \circ \gamma = [\mathsf{nt}_A(p_\kappa) \circ \alpha[a_{\kappa+1}] \circ F f_{\kappa+1}, \mathsf{nt}_A(p_\kappa) \circ \mathsf{nt}_A(a_{\kappa+1})] \circ \gamma = [\alpha[a_\kappa] \circ F(p_\kappa \circ f_{\kappa+1}), \mathsf{nt}_A(a_\kappa)] \circ \gamma = [\alpha[a_\kappa] \circ F f_\kappa, a_{\kappa+1}] \circ \gamma = f_{\kappa+1}$ from the induction hypothesis $p_\kappa \circ f_{\kappa+1} = f_\kappa$, using the fact $p_\kappa \circ a_{\kappa+1} = a_\kappa$, which implies $\mathsf{nt}_A(p_\kappa) \circ \alpha[a_{\kappa+1}] = \alpha[a_\kappa] \circ F p_\kappa$ and $\mathsf{nt}_A(p_\kappa) \circ \mathsf{nt}_A(a_{\kappa+1}) = \mathsf{nt}_A(a_\kappa)$ by the definition of the functorial extension of nt_A.

For κ a limit ordinal, $p_\kappa \circ f_{\kappa+1} = \langle p_\iota \circ \mathsf{nt}_A(p_{\kappa,\iota})\rangle_{\iota<\kappa} \circ [\alpha[a_\kappa] \circ F f_\kappa, a_{\kappa+1}] \circ \gamma = \langle p_\iota \circ \mathsf{nt}_A(p_{\kappa,\iota}) \circ [\alpha[a_\kappa] \circ F f_\kappa, a_{\kappa+1}] \circ \gamma\rangle_{\iota<\kappa} = \langle p_\iota \circ [\mathsf{nt}_A(p_{\kappa,\iota}) \circ \alpha[a_\kappa] \circ F f_\kappa, \mathsf{nt}_A(p_{\kappa,\iota}) \circ \mathsf{nt}_A(a_\kappa)] \circ \gamma\rangle_{\iota<\kappa} = \langle p_\iota \circ [\alpha[a_\iota] \circ F(p_{\kappa,\iota} \circ f_\iota), \mathsf{nt}_A(a_\iota)] \circ \gamma\rangle_{\iota<\kappa} = \langle p_\iota \circ [\alpha[a_\iota] \circ F f_\iota, a_{\iota+1}] \circ \gamma\rangle_{\iota<\kappa} = \langle p_\iota \circ f_{\iota+1}\rangle_{\iota<\kappa} = \langle f_\iota\rangle_{\iota<\kappa} = f_\kappa$ follows from the induction hypotheses $p_{\kappa,\iota} \circ f_\kappa = f_\iota$ and $p_\iota \circ f_{\iota+1} = f_\iota$, using the facts $p_{\kappa,\iota} \circ a_\kappa = a_\iota$, which imply $\mathsf{nt}_A(p_{\kappa,\iota}) \circ \alpha[a_\kappa] = \alpha[a_\iota] \circ F p_{\kappa,\iota}$ and $\mathsf{nt}_A(p_{\kappa,\iota}) \circ \mathsf{nt}_A(a_\kappa) = \mathsf{nt}_A(a_\iota)$ by the definition of the functorial extension of nt_A.

For κ a limit ordinal and $\iota < \kappa$, it is straightforward that $p_{\kappa,\iota} \circ f_\kappa = \pi_{\kappa,\iota} \circ \langle f_\iota\rangle_{\iota<\kappa} = f_\iota$.

Given that $A_\lambda = A_{\lambda+1} = A$, which implies that $p_\lambda = \mathsf{id}_A$, $a_{\lambda+1} = \mathsf{id}_A$, $\alpha[a_\lambda] = \alpha$, it is immediate that f_λ is a solution (in f) of the equation

$$\begin{array}{ccc} C & \xrightarrow{\gamma} & FC + A \\ {\scriptstyle f}\downarrow & & \downarrow{\scriptstyle Ff + \mathsf{id}_A} \\ A & \xleftarrow[{[\alpha, \mathsf{id}_A]}]{} & FA + A \end{array}$$

Indeed, $f_\lambda = p_\lambda \circ f_{\lambda+1} = f_{\lambda+1} = [\alpha[a_\lambda] \circ F f_\lambda, a_{\lambda+1}] \circ \gamma = [\alpha \circ F f_\lambda, \mathsf{id}_A] \circ \gamma$.

To show that it is the only solution, i.e., that, for any other solution f, we have $f = f_\lambda$, we show that $a_\kappa \circ f = f_\kappa$. We do this by induction.

The base case $a_0 \circ f = !_A \circ f = !_C = f_0$ holds trivially.

We also have $a_{\kappa+1} \circ f = \mathsf{nt}_A(a_\kappa) \circ f = \mathsf{nt}_A(a_\kappa) \circ [\alpha \circ F f, \mathsf{id}_A] \circ \gamma = [\mathsf{nt}_A(a_\kappa) \circ \alpha \circ F f, \mathsf{nt}_A(a_\kappa)] \circ \gamma = [\alpha[a_\kappa] \circ F(a_\kappa \circ f), \mathsf{nt}_A(a_\kappa)] \circ \gamma = [\alpha[a_\kappa] \circ F f_\kappa, a_{\kappa+1}] \circ \gamma = f_{\kappa+1}$, from the induction hypothesis $a_\kappa \circ f = f_\kappa$, using that f is a solution.

For κ a limit ordinal, we get $a_\kappa \circ f = \langle a_\iota\rangle_{\iota<\kappa} \circ f = \langle a_\iota \circ f\rangle_{\iota<\kappa} = \langle f_\iota\rangle_{\iota<\kappa} = f_\kappa$ from the induction hypotheses $a_\iota \circ f = f_\iota$ (for $\iota < \kappa$).

From this basis, the desired result $f = f_\lambda$ is already immediate: as $a_\lambda = \mathsf{id}_A$, it is trivial that $f = a_\lambda \circ f = f_\lambda$. □

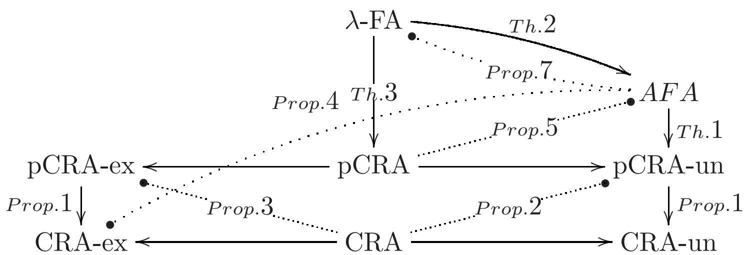
Finally, notice that since pCRA does not imply AFA, it cannot imply λ-FA. Example 6 shows this: We proved that it satisfies pCRA but not AFA, hence it cannot satisfy λ-FA either.

5 Conclusion

We have looked at some notions of support for general structured corecursion/coinduction. They are all properties on an algebra (A, α) of a fixed functor

F. The conditions CRA/pCRA state that we can uniquely solve all structured recursive diagrams based on (A, α). The condition AFA asserts that the principle of bisimilarity holds for the carrier A: Every equivalence on A that is finer than its own structural refinement must be equality. Finally, λ-FA says that we can reconstruct A by iterating structural refinement.

The relations between the four conditions CRA, pCRA, AFA, and λ-FA are summarized by the following diagram. The solid lines indicate implications, the dotted lines indicate non-implications.



We conclude from this study that general structured corecursion/coinduction is more subtle and, at the same time, also more revealing than general structured recursion/induction from which we drew inspiration. In particular, we have seen that, for Set-like categories, straightforward dualization of the different equivalent conditions of recursion/induction leads to inequivalent conditions of corecursion/coinduction. This could be an indication that some of the conditions are not really the right ones: perhaps they work for recursion/induction in Set incidentally, but for smooth generalization to other categories and dualization one should proceed from different conditions. While we believe firmly that recursiveness [corecursiveness] are natural conditions, it may turn out that some yet unconsidered versions of wellfoundedness [antifoundedness] are more robustly equivalent to recursiveness [corecursiveness] than the versions we have considered here.

To achieve progress we must fully understand each of the conditions we have considered and the role that the different viable assumptions play for the implications and non-implications between them. We can then seek variants that are more in line with our intuitive grasp. We expect that this enquiry will produce new and exciting results.

We would like to be able to tell a type-theoretic version of the story, i.e., to develop a dual Bove-Capretta method (allowing a general corecursive definition to be justified by a productivity proof). For this, we must overcome the discrepancies already commented, but likewise it is important that all our constructions can be made constructively (computationally) meaningful.

We would also very much like to relate our work to approaches to recursion/corecursion based on Banach's fixed point theorem [7,3].

Acknowledgements. We thank Thierry Coquand and Peter Dybjer for comments and questions and likewise the referees. T. Uustalu and V. Vene were supported by the Estonian Science Foundation under grants No. 6940 and 6713. The research visits between the authors were financed by the EU FP6 IST coordination action TYPES and the Estonian Tiger University Plus programme. The latter programme also financed Varmo Vene's attendance of SBMF 2009.

References

1. Adámek, J., Lücke, D., Milius, S.: Recursive coalgebras for finitary functors. Theor. Inform. and Appl. 41(4), 447–462 (2007)
2. Bove, A., Capretta, V.: Modelling general recursion in type theory. Math. Struct. in Comput. Sci. 14(4), 671–708 (2005)
3. Buchholz, W.: A term calculus for (co-)recursive definitions on streamlike data structures. Ann. of Pure and Appl. Logic 136(1–2), 75–90 (2005)
4. Capretta, V., Uustalu, T., Vene, V.: Recursive coalgebras from comonads. Inform. and Comput. 204(4), 437–468 (2006)
5. Cockett, R., Fukushima, T.: About Charity. Yellow series report 92/480/18, Dept. of Comput. Sci., Univ. of Calgary (1992)
6. Coquand, T.: Infinite objects in type theory. In: Barendregt, H., Nipkow, T. (eds.) TYPES 1993. LNCS, vol. 806, pp. 62–78. Springer, Heidelberg (1994)
7. Di Gianantonio, P., Miculan, M.: A unifying approach to recursive and co-recursive definitions. In: Geuvers, H., Wiedijk, F. (eds.) TYPES 2002. LNCS, vol. 2646, pp. 148–161. Springer, Heidelberg (2003)
8. Doornbos, H., Backhouse, R.: Reductivity. Sci. of Comput. Program 26(1-3), 217–236 (1996)
9. Eppendahl, A.: Coalgebra-to-algebra morphisms. Electron. Notes in Theor. Comput. Sci. 29, 8 (1999)
10. Giménez, E.: Codifying guarded definitions with recursion schemes. In: Smith, J., Dybjer, P., Nordström, B. (eds.) TYPES 1994. LNCS, vol. 996, pp. 39–59. Springer, Heidelberg (1995)
11. Jacobs, B.: The temporal logic of coalgebras via Galois algebras. Math. Struct. in Comput. Sci. 12(6), 875–903 (2002)
12. Milius, S.: Completely iterative algebras and completely iterative monads. Inform. and Comput. 196(1), 1–41 (2005)
13. Osius, G.: Categorical set theory: a characterization of the category of sets. J. of Pure and Appl. Algebra 4, 79–119 (1974)
14. Taylor, P.: Intuitionistic sets and ordinals. J. of Symb. Logic 61(3), 705–744 (1996)
15. Taylor, P.: Towards a unified treatment of induction, I: The general recursion theorem. Unpublished draft 35 pp (1996); // A short version (1996) of 5 pp was distributed at Gödel 1996 Brno (August 1996), `http://www.monad.me.uk/ordinals/`
16. Taylor, P.: Practical foundations of mathematics. Cambridge Studies in Advanced Mathematics, vol. 59, xi+572. Cambridge Univ. Press, Cambridge (1999)
17. Turner, D.A.: Total functional programming. J. of Univ. Comput. Sci. 10(7), 751–768 (2004)
18. Uustalu, T., Vene, V.: Primitive (co)recursion and course-of-value (co)iteration, categorically. Informatica 10(1), 5–26 (1999)