Initial Algebras Unchained

A Novel Initial Algebra Construction Formalized in Agda

Thorsten Wißmann
thorsten.wissmann@fau.de
Friedrich-Alexander-Universität Erlangen-Nürnberg
Erlangen, Germany

Stefan Milius*
stefan.milius@fau.de
Friedrich-Alexander-Universität Erlangen-Nürnberg
Erlangen, Germany

ABSTRACT

The initial algebra for an endofunctor F provides a recursion and induction scheme for data structures whose constructors are described by F. The initial-algebra construction by Adámek (1974) starts with the initial object (e.g. the empty set) and successively applies the functor until a fixed point is reached, an idea inspired by Kleene's fixed point theorem. Depending on the functor of interest, this may require transfinitely many steps indexed by ordinal numbers until termination.

We provide a new initial algebra construction which is not based on an ordinal-indexed chain. Instead, our construction is loosely inspired by Pataraia's fixed point theorem and forms the colimit of all finite recursive coalgebras. This is reminiscent of the construction of the rational fixed point of an endofunctor that forms the colimit of *all* finite coalgebras. For our main correctness theorem, we assume the given endofunctor is accessible on a (weak form of) locally presentable category. Our proofs are constructive and fully formalized in Agda.

CCS CONCEPTS

• **Theory of computation → Logic**.

KEYWORDS

Agda, Initial Algebra, Recursive Coalgebra, Presentable Category

ACM Reference Format:
Thorsten Wißmann and Stefan Milius. 2024. Initial Algebras Unchained: A Novel Initial Algebra Construction Formalized in Agda. In *39th Annual ACM/IEEE Symposium on Logic in Computer Science (LICS '24), July 8–11, 2024, Tallinn, Estonia.* ACM, New York, NY, USA, 14 pages. https://doi.org/10.1145/3661814.3662105

1 INTRODUCTION

Structural recursion is a fundamental principle in computer science; it appears whenever one traverses syntax or other inductively defined data structures such as natural numbers, lists, or trees. Any concrete recursion principle depends on the type of the constructors of the syntax or data structure of interest. A uniform view on these recursion principles is provided by the theory of algebras for an endofunctor F on a category, where F is a parameter modelling the type of constructors. An algebra for F can be understood as an object of data with operations of the type modelled by F.

An initial algebra for F – if it exists – provides the data structure which is the canonical minimal implementation of a data type with constructors specified by F. Its universal property yields a recursion and an induction principle for this data structure.

For example, for the set functor F defined by $FX = \{\bullet\} + X \times X$ the initial F-algebra (I, i) consists of all binary trees. A standard example of a recursively defined function on binary trees is the *height*-function $h \colon I \to \mathbb{N}$ defined by

$$h(\bullet) = 0 \qquad \text{and} \qquad h\Big(\overset{\wedge}{s \quad t}\Big) = 1 + \max(h(s), h(t)), \qquad (1)$$

where the second case considers the binary tree obtained by joining the trees $s, t \in I$ under a new root node. Initiality means that for every F-algebra there exists a unique homomorphism from (I, i) to (A, a). For instance, the height-function h is the unique function from the initial algebra (I, i) to the F-algebra on $A = \mathbb{N}$ with the following structure a (see also Figure 1):

$$a \colon F\mathbb{N} \to \mathbb{N} \quad a(\mathsf{inl}(\bullet)) = 0 \quad a(\mathsf{inr}(k, n)) = 1 + \max(k, n); \qquad (2)$$

here $\mathsf{inl} \colon \{\bullet\} \to \{\bullet\} + \mathbb{N} \times \mathbb{N}$ and $\mathsf{inr} \colon \mathbb{N} \times N \to F\mathbb{N}$ denote the coproduct injections. This models that a leaf has height 0, and an inner node with children of height k and n, respectively, has a height of $1+\max(k, n)$. Note that the commutativity of the left-hand square in Figure 1 is equivalent to the two equations in (1); in particular, one sees that the arguments k, n of the algebra structure a can be thought of as the returned values of the recursive calls of h to the maximal subtrees s and t of a given binary tree which is not simply a leaf.

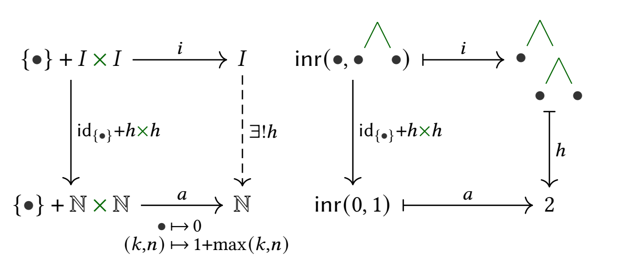


Figure 1: Example of an algebra $(\mathbb{N}, a)$ and the algebra homomorphism h induced by the initial algebra (I, i)

*Funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) – project numbers 470467389 and 517924115

LICS '24, July 8–11, 2024, Tallinn, Estonia

ACM ISBN 979-8-4007-0660-8/24/07
https://doi.org/10.1145/3661814.3662105

Since initial algebras are an important concept, it is natural to investigate, when initial algebras exist and how they are constructed. Their existence is entailed by assumptions on the 'smallness' of the constructions performed by the endofunctor F. More precisely, every accessible endofunctor on a locally presentable category has an initial algebra; this follows from classical results by Adámek [5]. He also provided a construction of the initial algebra for an endofunctor F generalizing Kleene's construction of the least fixed point of a continuous function on a cpo (or more generally, the corresponding transfinite version for monotone maps on chain-complete posets, which is essentially due to Zermelo [27]). For the category of sets, the construction builds the *initial-algebra chain*, an ascending chain of sets, by transfinite recursion. It starts with the initial object, the empty set, and then successively applies the functor F:

$$\emptyset \to F\emptyset \to F^2\emptyset \to F^3\emptyset \to \cdots$$

At a limit ordinal i, one takes the colimit of the chain of all previous sets. If the initial-algebra chain converges in the sense that a connecting map from one step to the next is an isomorphism, then its inverse is the structure of an initial algebra for F.

This provides a nice iterative construction of initial algebras. However, the use of transfinite induction makes it difficult to formalize this classical initial-algebra construction in full generality in a proof assistant based on type theory, such as Agda, Coq, or Lean. The problem seems to be that the notion of an ordinal is inherently set-theoretic, and therefore does not easily translate into type theory. It is the goal of our paper to provide a new construction of the initial algebra which does not rely on transfinite recursion and can be formalized in proof assistants.

Our construction is based on coalgebras obeying the recursion scheme [24], aka. *recursive coalgebras* [9, 12, 20]. They are coalgebras $r\colon R \to FR$ satisfying a property much like the universal property of an initial algebra: for every F-algebra $b\colon FB \to B$, there exists be a unique coalgebra-to-algebra morphism:

$$\begin{array}{ccc} FR & \xleftarrow{\;r\;} & R \\ {\scriptstyle Fh}\downarrow & & \downarrow{\scriptstyle \exists! h} \\ FB & \xrightarrow{\;b\;} & B \end{array}$$

Recursive coalgebras have originally arisen in the categorical study of well-founded induction [20]; in fact, under mild assumptions on the endofunctor F a coalgebra is recursive iff it is *well-founded* [2, 15, 23–25]; the latter notion generalizes the classical notion of a well-founded relation to the level of coalgebras for an endofunctor.

Consequently, a recursive coalgebra $R \to FR$ models a kind of decomposing or 'divide' step within a recursive (divide-and-conquer) computation; this has been nicely explained by Capretta et al. [9]. Since it does not need be closed under operations of type F, a recursive coalgebra allows us to collect only some of the inhabitants from the initial algebra or graph-like versions of the syntactic tree-like elements, e.g. sharing subtrees. For example, Figure 2 shows an example of a recursive coalgebra for the set functor $FX = \{\bullet\} + X \times X$. This coalgebra is indeed recursive,

R	r	FR
u	$\mapsto$	$\mathsf{inr}(x, x)$
v	$\mapsto$	$\mathsf{inr}(y, w)$
w	$\mapsto$	$\mathsf{inr}(z, y)$
x	$\mapsto$	$\mathsf{inl}(\bullet)$
y	$\mapsto$	$\mathsf{inl}(\bullet)$
z	$\mapsto$	$\mathsf{inl}(\bullet)$

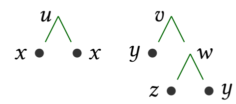


Figure 2: Example of a recursive F-coalgebra $r\colon R \to FR$

because for every algebra $b\colon FB \to B$, the following function

$$\begin{aligned} h\colon R &\longrightarrow B \\ h(x) := h(y) := h(z) &:= b(\mathsf{inl}(\bullet)) \\ h(u) &:= b(\mathsf{inr}(h(x), h(x))) \\ h(w) &:= b(\mathsf{inr}(h(z), h(y))) \\ h(v) &:= b(\mathsf{inr}(h(y), h(w))) \end{aligned}$$

specifies a unique coalgebra-to-algebra morphism from (R, r) to (B, b). So recursive coalgebras can intuitively be understood as well-founded data objects whose constructor types are specified by F.

An important special case is that if the structure $r\colon R \to FR$ of a recursive coalgebra happens to be an isomorphism, then (R, r^{-1}) is the initial algebra.

1.1 Overview of the Contribution

In the present work, we use this observation to provide a new construction of the initial algebra. Intuitively, our construction is based on the idea that

> *The initial algebra is the collection of well-founded data object of type F (modulo behavioural equivalence).*

Our construction works for every accessible endofunctor on a locally presentable category. To make the point that our proof is constructive, we formalize it in Agda. To this end we use the notion of a Fil-accessible category, for a collection Fil of filtered colimits, which subsumes the notion of a locally λ-presentable category (for a regular cardinal λ), but without the need to mention any cardinal number λ explicitly.

To construct an initial algebra, we consider recursive coalgebras $r\colon R \to FR$ where R is a *presentable* object; this notion generalizes that of a finite set. Therefore, such coalgebras are said to be *finite-recursive*. We then take the colimit of all finite-recursive coalgebras for F and obtain a coalgebra $\alpha\colon A \to FA$. Even though A is not finitely presentable, (A, α) is *locally* finite-recursive, in the sense that it is built as a colimit from finite-recursive coalgebras. We prove that (A, α) satisfies a universal property: it is the *terminal* locally finite-recursive coalgebra.

Finally, we prove that it is a fixed point of F. To this end, we provide a non-trivial argument showing that $(FA, F\alpha)$ is also locally finite-recursive. An argument in the style of Lambek's Lemma then shows that α is an isomorphism, whence (A, α^{-1}) is the initial algebra.

Structure of this paper. After discussing categorical preliminaries on locally presentable categories and recursive coalgebras (Section 2), we establish our main initial algebra theorem (Section 3). The relation and differences to the construction based on the initial-algebra chain are discussed in Section 4. We address technical aspects and conceptual challenges of the Agda formalization in Section 5. The main text concludes in Section 6. The index of formalized results (Section 7) links the results from the paper with the corresponding result in Agda.

1.2 Agda Formalization

Our theorem is fully formalized in Agda (2.6.4) using the `agda-categories` library (v0.2.0) [14]. Despite the effort, we went for the formalization for the following reasons:

(1) Underpin our claim that our results are constructive. Agda makes it explicitly visible where non-constructive methods (such as the law of excluded middle) are used in our proofs or in proofs of standard lemmas (especially on hom-colimits and locally presentable categories). In addition, the distinction between small and large colimits becomes visible due to Agda's type level system.

(2) Exclude mistakes; especially the proof that $(FA, F\alpha)$ is locally finite-recursive turned out to be non-trivial.

(3) Contribute to the growing field of mechanized mathematics. We kept many lemmas on colimits and coalgebras as general as possible and will submit them to the `agda-categories` project.

The paper is phrased in standard set theory and category theory to be more accessible to general readers, but we keep it as close to our Agda formalization as possible.

Our Agda source code spans more than 5000 lines and 29 files. Both the source code and the HTML documentation can be found in the ancillary files on arxiv.org and on:

https://git8.cs.fau.de/software/initial-algebras-unchained
(also archived on archive.softwareheritage.org)

We annotate mechanized definitions and theorems with a clickable icon which links to the online HTML documentation of the respective result in the Agda code base. So readers who want to have a quick look may browse the linked HTML documentation. Additional clues about the formalization (and clickable links) can be found in the index in Section 7 at the end of this document.

1.3 Related Work

Adámek et al. [3] have provided a proof of an initial algebra theorem which uses recursive coalgebras to construct the initial algebra from a given pre-fixed point of F (that is, an algebra A whose structure morphism is monomorphic). They use Pataraia's fixed point theorem to obtain the initial algebra as the least fixed point of a monotone operator on the lattice of subobjects of A. However, assuming existence of a pre-fixed point is quite strong; establishing this is almost as difficult as proving existence of an initial algebra. Our construction works without that assumption and obtains (the carrier of) the initial algebra as a colimit.

Pitts and Steenkamp [21] formalized an initial algebra theorem in Agda using *inflationary iteration* and avoiding transfinite iteration.

There have been efforts to incorporate ordinals in Agda [13] and transfinite induction in Coq [22]. However, it is not clear to us whether this could be used for any formalization of the initial-algebra chain or the notion of a locally λ-presentable category.

Our construction is reminiscent of the construction of the rational fixed point of a finitary endofunctor on a locally finitely presentable category [4]. This is constructed by taking the colimit of *all* coalgebras with a finitely presentable carrier (in lieu of all finite-recursive) coalgebras. In addition, our technique of applying the functor to a locally finite-recursive coalgebra (Theorem 3.22) is reminiscent of what is done in work on the rational fixed point [17–19, 26] to show that it is indeed a fixed point. Beside incorporating recursiveness, we needed to slightly adapt the ideas to make them work in the Agda formalization.

Our notion of a Fil-accessible category (Definition 3.1) is very similar to a notion used by Urbat [26]. It is also reminiscent of the notion of a $\mathbb{D}$-accessible category introduced by Adámek et al. [1] and further studied by Centazzo et al. [10, 11].

2 CATEGORICAL PRELIMINARIES

We assume basic knowledge of category theory, functors, and colimits; see [6, 8] for a detailed introduction.

Notation 2.1. Both in this paper and in the Agda source code, we use calligraphic letters $\mathscr{C}, \mathscr{D}, \mathscr{E}$ for categories, latin letters for functors and objects, and serif-free font for identifiers consisting of multiple letters, such as Set for the category of sets and maps.

We denote the coproduct of objects $X, Y \in \mathscr{C}$ by $X + Y$ (if it exists) and we write $[f, g]: X + Y \to Z$ for the unique morphism induced by the morphisms $f: X \to Z$ and $g: Y \to Z$.

2.1 Finiteness in a category

A standard way to capture the notion of finiteness of an object in a category is in terms of the preservation of filtered colimits:

Definition 2.2. We recall from Adámek and Rosický [7]:

(1) A functor $F: \mathscr{C} \to \mathscr{C}'$ *preserves the colimit* $(c_i: Di \to C)_{i\in\mathscr{D}}$ of a diagram $D: \mathscr{D} \to \mathscr{C}$ if $(Fc_i: FDi \to FC)_{i\in\mathscr{D}}$ is a colimit of the diagram $F \circ D: \mathscr{D} \to \mathscr{C}'$.

(2) A category $\mathscr{D}$ is *filtered* provided that
  (a) $\mathscr{D}$ is non-empty,
  (b) for every $X, Y \in \mathscr{D}$ there is an *upper bound* $Z \in \mathscr{D}$, that is, there are morphisms $X \to Z$ and $Y \to Z$,
  (c) for every $f, g: X \to Y$ there is some $Z \in \mathscr{D}$ and some $h: Y \to Z$ with $h \circ g = h \circ f$.

A diagram $D: \mathscr{D} \to \mathscr{C}$ is *filtered* if $\mathscr{D}$ is a filtered category, and a colimit of a filtered diagram is said to be *filtered*.

(3) A functor $F: \mathscr{C} \to \mathscr{C}'$ is *finitary* if it preserves filtered colimits.

(4) An object $X \in \mathscr{C}$ is *finitely presentable* if its hom-functor $\mathscr{C}(X, -): \mathscr{C} \to \mathsf{Set}$ is finitary.

Remark 2.3. Note that the notions of small and large diagram schemes behave slightly differently in Agda than in classic set theory when quantifying over all sets. Therefore, we do not distinguish between small and large diagram schemes $\mathscr{D}$ above: preservation of colimits simply refers to those colimits that exist.

Remark 2.4. Finitariness can equivalently be defined using *directed* diagrams (i.e. those where the diagram scheme is a directed poset) in lieu of filtered ones [7, Thm. 1.5]. We decided to work with filtered diagrams because this simplifies the formalization of locally finitely presentable categories (Definition 2.9).

Example 2.5 [7, Ex. 1.2]**.** Finite presentability instantiates to standard notions of finiteness in the following categories:

(1) In the categories of sets, posets and graphs, the finitely presentable objects are precisely the finite sets, posets and graphs, respectively.

(2) In the category of vector spaces over a field k, the finitely presentable objects are precisely the finite dimensional vector spaces.

(3) In the category of groups and monoids, the finitely presentable objects are precisely those which can be presented by finitely many generators and relations.

(4) More generally, in every finitary variety, i.e. a category of algebras for a finitary signature satisfying a set of equations, the finitely presentable objects are precisely those presented by finitely many generators and relations.

Remark 2.6. Filtered colimits in Set can be characterized as follows. Given a filtered diagram $D\colon \mathscr{D} \to \mathsf{Set}$, a cocone $c_i\colon Di \to C$ ($i \in \mathscr{D}$) is a colimit of D if and only if it satisfies the two conditions:

(1) the colimit injections are jointly surjective: for every $x \in C$ there exist $i \in \mathscr{D}$ and $x' \in Di$ such that $x = c_i(x')$;

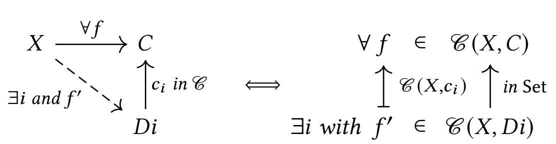
(2) for every pair $x', x'' \in Di$ with $c_i(x') = c_i(x'')$ there exists a morphism $h\colon i \to j$ in $\mathscr{D}$ such that $Dh(x') = Dh(x'')$.

General (non-filtered) colimits in Set have a similar characterization given by condition (1) and a more involved version of (2). Condition item (2) makes use of filteredness: whenever two elements are identified in the colimit, then are already identified by a connecting morphism of the diagram.

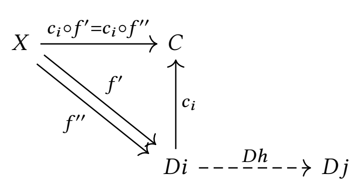
When instantiating this characterization to filtered diagrams $D\colon \mathscr{D} \to \mathscr{C}$ postcomposed with a hom-functor $\mathscr{C}(X, -)$ of some $X \in \mathscr{C}$, we obtain:

Lemma 2.7 (✓)**.** *The hom-functor $\mathscr{C}(X, -)$ for an object X preserves the colimit C of a filtered diagram D iff every morphism $f\colon X \to C$ factorizes essentially uniquely through one of the colimit injection $c_i\colon Di \to C$ ($i \in \mathscr{D}$) in the sense that:*

(1) *there exist $i \in \mathscr{D}$ and $f'\colon X \to Di$ such that $f = c_i \circ f'$:*



(2) *given two such factorizations $c_i \circ f' = c_i \circ f''$ of f there exists a morphism $h\colon i \to j$ of $\mathscr{D}$ such that $Dh \circ f' = Dh \circ f''$.*



Indeed, apply Remark 2.6 to the cocone $\mathscr{C}(X, c_i)$ ($i \in \mathscr{D}$).

Next we recall the notion of a locally finitely presentable category. The idea is that every object in such a category can be constructed from a set of finitely presentable ones in a canonical way:

Definition 2.8 (✓)**.** For a set $\mathcal{S}$ of objects of $\mathscr{C}$ and $X \in \mathscr{C}$, define the category $\mathcal{S}/X$ to have

- objects (S, f) for $S \in \mathcal{S}$ and $f\colon S \to X$ (in $\mathscr{C}$), and
- morphisms $h\colon (S, f) \to (T, g)$ for $h\colon S \to T$ with $g \circ h = f$ in $\mathscr{C}$.

The functor $U_X\colon \mathcal{S}/X \to \mathscr{C}$ is defined by $U_X(S, f) = S$. Every such diagram U_X has a canonical cocone:

$$U_X(S, f) \xrightarrow{f} X \qquad \text{for } (S, f) \in \mathcal{S}/X. \tag{3}$$

Definition 2.9 [7]**.** A category $\mathscr{C}$ is *locally finitely presentable* (*lfp*, for short) provided that it is cocomplete and has a set $\mathscr{C}_\mathrm{p}$ of finitely presentable objects such that every object $X \in \mathscr{C}$ is the colimit of $U_X\colon \mathscr{C}_\mathrm{p}/X \to \mathscr{C}$.

Example 2.10. Examples of lfp categories are ubiquitous. For example, the categories of sets, posets and graphs are lfp. Every finitary varieties of algebras forms an lfp category. Instances of this are the categories of groups, monoids and vector spaces and many others.

2.2 Algebra and Coalgebra

We recall some basic notions from the theory of (co-)algebras for an endofunctor.

Definition 2.11. Given an endofunctor $F\colon \mathscr{C} \to \mathscr{C}$, an F-algebra is a pair (A, a) where A is an object of $\mathscr{C}$ (the *carrier* of the algebra) and with a morphism $a\colon FA \to A$ a morphism (its *structure*). Dually, an F-coalgebra is a pair (C, c) consisting of a carrier object C and a structure morphism $c\colon C \to FC$.

Intuitively, an F-algebra provides an object of data values together with operations described by F. For example, for the set functor $FX = \{\bullet\} + X \times X$, an F-algebra $a\colon FA \to A$ consists of

- some constant $a(\mathsf{inl}(\bullet)) \in A$, and
- a binary operation sending $x, y \in A$ to $a(\mathsf{inr}(x, y)) \in A$.

An F-coalgebra $c\colon C \to FC$ provides an object of states with abstract transitions or a structured collection of successors described by F. For example, for the above set functor F we have for every state $x \in C$ that

- either $c(x) \in \{\bullet\}$, describing that x has no successor,
- or $c(x) \in C \times C$, describing that x has a pair of successors.

So coalgebras can be thought of as graph like structures.

We relate F-algebras and F-coalgebras using the following notions of morphisms:

Definition 2.12. An *F-algebra morphism* h from (A, a) to (B, b) is a $\mathscr{C}$-morphism $h\colon A \to B$ satisfying $h \circ a = b \circ Fh$ (see the right-hand square in diagram (4)). An *F-coalgebra morphism* g from (C, c) to (D, d) is a $\mathscr{C}$-morphism $g\colon C \to D$ satisfying $d \circ g = Fg \circ c$ (see the left-hand square in (4)). A *coalgebra-to-algebra morphism* from a coalgebra (D, d) to an algebra (A, a) is a $\mathscr{C}$-morphism $s\colon D \to A$

such that $s = a \circ Fs \circ d$ (see the middle square in (4)).

$$\begin{array}{ccccccc} C & \xrightarrow{g} & D & \xrightarrow{s} & A & \xrightarrow{h} & B \\ {\scriptstyle c}\downarrow & & {\scriptstyle d}\downarrow & & \uparrow{\scriptstyle a} & & \uparrow{\scriptstyle b} \\ FC & \xrightarrow{Fg} & FD & \xrightarrow{Fs} & FA & \xrightarrow{Fh} & FB \end{array} \tag{4}$$

Definition 2.13. A coalgebra (R, r) is *recursive* if for every algebra (A, a) there is a unique coalgebra-to-algebra morphism from (R, r) to (A, a).

In addition to the example of a recursive coalgebra for $FX = \{\bullet\} + X \times X$ in the introduction, we now discuss further examples which illustrate the point that recursive coalgebras relate to well-founded induction and that they capture the 'divide' step in recursive divide-and-conquer computations.

Example 2.14. (1) The first examples of recursive coalgebras are well-founded relations. Recall that a binary relation R on a set X is well-founded iff there is no infinite descending sequence of related elements:

$$\cdots R\, x_3\; R\, x_2\; R\, x_1\; R\, x_0.$$

A binary relation $R \subseteq X \times X$ is, equivalently, a coalgebra for the power-set functor $\mathscr{P}$ which maps a set X to the set $\mathscr{P}X$ of all its subsets: R corresponds to $c\colon X \to \mathscr{P}X$ defined by $c(x) = \{y : y\, R\, x\}$. The relation R is well-founded iff the associated $\mathscr{P}$-coalgebra is recursive.

(2) For every endofunctor F having an initial algebra $i\colon FI \to I$, its structure is an isomorphism (by Lambek's Lemma [16]), and the inverse $i^{-1}\colon I \to FI$ is a recursive coalgebra.

(3) Capretta et al. [9] have shown how to obtain Quicksort using recursivity. Let C be any linearly ordered set (of data elements). Quicksort is the recursive function $q\colon C^* \to C^*$ defined by

$$q(\varepsilon) = \varepsilon \qquad \text{and} \qquad q(cw) = q(w_{\le c}) \star (cq(w_{>c})),$$

where C^* is the set of all lists on C, ε is the empty list, $\star$ is the concatenation of lists and $w_{\le c}$ denotes the list of those elements of w that are less than or equal to $c \in C$; analogously for $w_{>c}$.

Now consider the set functor $FX = \{\bullet\} + C \times X \times X$ and define the coalgebra $s\colon C^* \to \{\bullet\} + C \times C^* \times C^*$ by

$$s(\varepsilon) = \bullet \quad \text{and} \quad s(cw) = (c, w_{\le c}, w_{>c}) \quad \text{for } c \in C \text{ and } w \in C^*.$$

This coalgebra is recursive. Thus, for the following F-algebra $m\colon \{\bullet\} + C \times C^* \times C^* \to C^*$ defined by

$$m(\bullet) = \varepsilon \qquad \text{and} \qquad m(c, w, v) = w \star (cv)$$

there exists a unique function q on C^* such that $q = m \circ Fq \circ s$.

Note that this equation reflects the idea that Quicksort is a divide-and-conquer algorithm. The coalgebra structure s divides a list into two parts $w_{\le c}$ and $w_{>c}$. Then Fq sorts these two smaller lists, and finally, in the combine (or conquer) step, the algebra structure m merges the two sorted parts to obtain the desired whole sorted list.

Jeannin et al. [15, Sec. 4] provide a number of further recursive functions arising in programming that are determined by recursivity of a coalgebra, e.g. the Euclidean algorithm for the gcd of integers, Ackermann's function, and the Towers of Hanoi.

Notation 2.15. We denote the category of F-coalgebras with their morphisms by F-Coalg and the respective category of F-algebras by F-Alg. The canonical forgetful functor $V\colon F\text{-Coalg} \to \mathscr{C}$ is defined by $V(C, c) = C$ on objects, and it is identity on morphisms.

Lemma 2.16 (). *The forgetful functor $V\colon F$-Coalg $\to \mathscr{C}$ creates all colimits.*

This means that, given a diagram $D\colon \mathscr{D} \to F$-Coalg, if the composed diagram $V \circ D\colon \mathscr{D} \to \mathscr{C}$ has a colimit $(h_i\colon VDi \to C)_{i \in \mathscr{D}}$, then there is a unique coalgebra structure $c\colon C \to FC$ such that all the colimit injections h_i are coalgebra morphisms. Moreover, (C, c) is then a colimit of D.

Lemma 2.17 (). *Recursive coalgebras are closed under colimits created by V.*

Proofsketch. Suppose that $D\colon \mathscr{D} \to F$-Coalg is a diagram such that $Dd = (C_d, c_d)$ is recursive for every $d \in \mathscr{D}$, we have to show that its colimit (C, c) is recursive, too. We denote the colimit injections by $i_d\colon (C_d, c_c) \to (C, c)$.

Given an algebra (A, a), we obtain for every $d \in D$ a unique coalgebra-to-algebra morphism h_d from (C, d) to (A, a). It is not difficult to prove that (h_d) is a cocone (of VD). Since the colimit of D is created by V, there exists a unique morphism $h\colon C \to A$ such that $h \circ i_d = h_d$ for all $d \in \mathscr{D}$. One now readily proves that h is a unique coalgebra-to-algebra morphism from (C, c) to (A, a). ◻

A helpful lemma to prove recursiveness is the following

Lemma 2.18 (, [9, Prop. 5]). *Consider coalgebra morphisms*

$$h\colon (R, r) \to (B, b) \textit{ and } g\colon (B, b) \to (FR, Fr)$$

with $b = Fh \circ g$. If (R, r) is recursive, then (B, b) is recursive, too.

For $g = \mathrm{id}$, we immediately obtain a corollary stating that recursive coalgebras are closed under application of F:

Corollary 2.19 (, [9, Prop. 6]). *If (R, r) is a recursive coalgebra, then so is (FR, Fr).*

Lemma 2.20 (, [9, Prop. 7(a)]). *If the structure $r\colon R \to FR$ of a recursive coalgebra is an isomorphism, then $r^{-1}\colon FR \to R$ is the initial algebra.*

3 INITIAL ALGEBRA THEOREM

Our main result (Theorem 3.31) states that for every accessible endofunctor F on a locally presentable category $\mathscr{C}$ the initial algebra can be constructed as the colimit of all sufficiently 'small' recursive coalgebras; 'small' here means that the carriers of those recursive coalgebra are λ-presentable, where λ is a regular cardinal such that $\mathscr{C}$ is locally λ-presentable and F is λ-accessible.

Since in our Agda formalization the treatment of the cardinal number λ is problematic we work in a slightly generalized setting:

Definition 3.1 (). Let Fil be a collection[1] of filtered categories.

(1) An object $X \in \mathscr{C}$ is (Fil-)*presentable* if its hom functor $\mathscr{C}(X, -)$ preserves colimits of diagrams $D\colon \mathscr{D} \to \mathscr{C}$ with $\mathscr{D} \in$ Fil.

(2) A category $\mathscr{C}$ is Fil-*accessible* provided that

(a) there is a set $\mathscr{C}_{\mathsf{p}} \subseteq \mathscr{C}$ of (Fil-)presentable objects,

(b) for all $X \in \mathscr{C}$, the slice category $(\mathscr{C}_{\mathsf{p}}/X)$ lies in Fil,

[1] In Agda, Fil is realized as a predicate; in classic set theory, it suffices to consider Fil as a class of small, filtered categories.

(c) every object $X \in \mathscr{C}$ is the colimit of $U_X\colon \mathscr{C}_\mathsf{p}/X \to \mathscr{C}$ (Definition 2.8).

Remark 3.2. (1) Definition 3.1 is very similar to an instance of an $(\mathbb{I}, \mathcal{M})$-accessible category in the sense of Urbat [26, Def. 3.4]. His notion is parametric in a collection $\mathbb{I}$ of sifted diagrams and the class $\mathcal{M}$ of an $(\mathcal{E}, \mathcal{M})$-factorization system which $\mathscr{C}$ is equipped with. Modulo the assumption on existence of colimits our notion is an instance of his by taking $\mathbb{I} = \mathsf{Fil}$ and the trivial factorization system (isomorphisms, all morphisms).

(2) Definition 3.1 is also reminiscent of the notion of a $\mathbb{D}$-accessible category introduced by Adámek et al. [1, Def. 3.4]. The parameter $\mathbb{D}$ is a *limit doctrine*, that is, an essentially small collection of small categories. They consider $\mathbb{D}$-filtered colimits, which are colimits that commute in Set with all limits with a diagram scheme in $\mathbb{D}$. For example, for $\mathbb{D}$ consisting of all finite categories, one obtains filtered colimits, and for $\mathbb{D}$ consisting of all finite discrete categories, one obtains sifted colimits. The precise relationship of this notion to ours (or Urbat's notion) is subject to further study.

(3) Unlike the classical definition of a locally presentable category (and the generalized notions in the previous two items) our definition explicitly mentions the canonical diagram U_X, and we explicitly require that $\mathscr{C}_\mathsf{p}/X$ lies in Fil. The reason for this is that our proofs use that the hom functor of a presentable object preserves the colimit in Definition 3.1.(2c).

Example 3.3. Every lfp category $\mathscr{C}$ is Fil-accessible for Fil being the class of all filtered categories.

Example 3.4. For the same collection Fil, the category Set is Fil-accessible, with $\mathscr{C}_\mathsf{p}$ being the category of all finite ordinals $\{0, \dots, k-1\}$ for $k \in \mathbb{N}$ and all maps between them.

We have chosen to work with Fil-accessible categories because their definition does not explicitly mention cardinal numbers. However, besides lfp categories, they subsume the more general notion of a locally λ-presentable category [7], for a regular cardinal λ:

Example 3.5. Let λ be a regular cardinal, and recall from op. cit. that a category $\mathscr{D}$ is *λ-filtered* if every set of morphsism of size less than λ has a cocone in $\mathscr{D}$. Equivalently:

(1) Every set of less than λ objects has a cocone, and

(2) every family of less than λ parallel morphisms $f_i\colon A \to B$ ($i \in I$ with $|I| < \lambda$) has a coequalizing morphism $g\colon B \to C$ (that is, $g \circ f_i\colon A \to C$ is independent of $i \in I$).

A diagram $D\colon \mathscr{D} \to \mathscr{C}$ is *λ-filtered* if so is its diagram scheme $\mathscr{D}$, and a *λ-filtered* colimit is a colimit of a λ-filtered diagram. A functor is *λ-accessible* if it preserves λ-filtered colimits. An object X of a category $\mathscr{C}$ is *λ-presentable* if its hom-functor $\mathscr{C}(X, -)$ is λ-accessible.

Finally, a category is *locally λ-presentable* if it is cocomplete and has a set of λ-presentable objects $\mathscr{C}_\mathsf{p}$ such that every object of $\mathscr{C}$ is a λ-filtered colimit of the diagram $U_X\colon \mathscr{C}_\mathsf{p}/X \to \mathscr{C}$ (cf. Definition 3.1).

For Fil being the class of all small λ-filtered categories, every locally λ-presentable category is Fil-accessible.

Note that like in lfp categories, the set $\mathscr{C}_\mathsf{p}$ does not necessarily need to contain all presentable objects (up to isomorphism):

Lemma 3.6. *Every presentable object $X \in \mathscr{C}$ is a split quotient of some object $P \in \mathscr{C}_\mathsf{p}$.*

That is, there exists a split epimorphism $P \twoheadrightarrow X$.

Assumption 3.7. For the remainder of this section, we fix a collection Fil of filtered categories, Fil-accessible category $\mathscr{C}$ and an endofunctor $F\colon \mathscr{C} \to \mathscr{C}$. We also assume that every pair X, Y of presentable objects has a coproduct $X + Y$.

It then follows that $X + Y$ is presentable, too.

Lemma 3.8. *Presentable objects are closed under binary coproduct in $\mathscr{C}$.*

Proofsketch. Similarly as in the setting of lfp categories, one uses filteredness (implied by Fil) in order to lift the desired factorization property from presentable objects X and Y to $X + Y$. ◻

Definition 3.9. A coalgebra (C, c) is

(1) *finite-recursive* (*finrec*) if it is recursive and C presentable.

(2) *locally finrec* if it is the colimit of finrec coalgebras. This means that there is a diagram $D\colon \mathscr{D} \to F\text{-Coalg}$ and a cocone $(Di \to (C, c))_{i \in \mathscr{D}}$ such that

(a) Di is finrec for every $i \in \mathscr{D}$,

(b) $(VDi\colon VDi \to C)_{i \in \mathscr{D}}$ is a colimit in $\mathscr{C}$ (where $V\colon F\text{-Coalg} \to \mathscr{C}$ is the forgetful functor).

Informally speaking, a finrec coalgebra is a system with transitions of type F that is free of cycles and in which every element has a finite description. This is precisely the intuition of the elements of the initial algebra. Consequently, our main theorem characterizes the initial algebra (considered as a coalgebra) as the colimit A of *all* finrec coalgebras. While we obtain the coalgebra structure $A \to FA$ directly from colimit creation (Lemma 2.16), we need some non-trivial results about locally finrec coalgebras in order to construct its inverse viz. the algebra structure $FA \to A$. Note that as soon as we have established an isomorphism $A \cong FA$, it is the initial algebra (by Lemma 2.20).

The way we obtain the desired isomorphism is very similar to the argument of Lambek's famous lemma [16]. In dual form this argument can be stated as follows:

Lemma 3.10. *The structure $c\colon C \to FC$ of a coalgebra (C, c) is an isomorphism provided that:*

(1) *there is a coalgebra morphism $h\colon (FC, Fc) \to (C, c)$,*

(2) *there is at most one coalgebra morphism $(C, c) \to (C, c)$.*

The Agda formalization of the proof is quite easy to read; in fact, this is a good place to start delving into our Agda code.

Proof. First note that the coalgebra structure is a coalgebra morphism $c\colon (C, c) \to (FC, Fc)$. Composing it with the coalgebra morphism $h\colon (FC, Fc) \to (C, c)$ according to item (1) yields an endomorphism on (C, c), which must be the identity by item (2): $h \circ c = \mathsf{id}_C$. In the following computation we use this, functoriality of F, and that h is a coalgebra morphism to conclude that it is the inverse of c:

$$c \circ h = Fh \circ Fc = F(h \circ c) = F\mathsf{id}_C = \mathsf{id}_{FC}. \qquad ◻$$

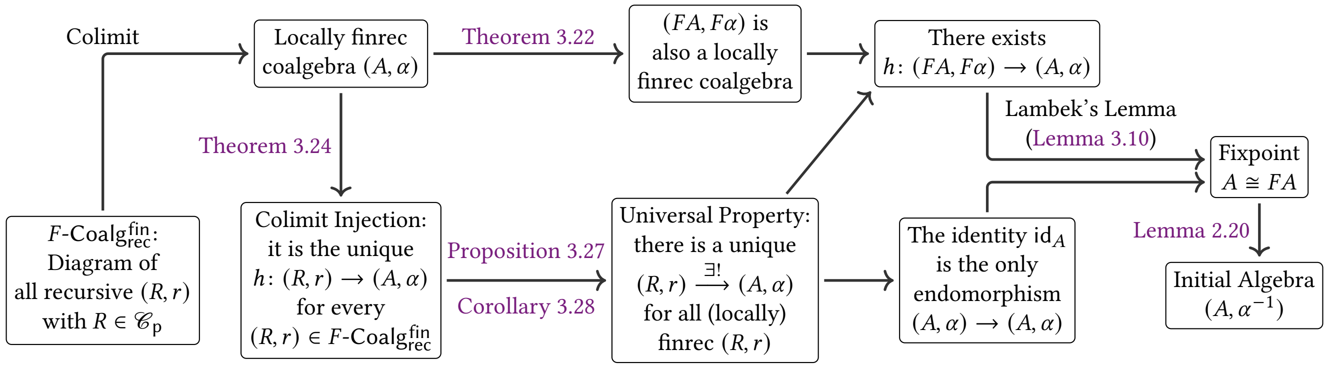


Figure 3: Roadmap for the initial algebra theorem (Theorem 3.30)

Remark 3.11. Note that the (dual of the) original Lemma assumes that (C, c) is terminal to establish the two conditions above.

In order to verify the two condition of Lemma 3.10 for our coalgebra $A \to FA$ we will develop elements of the theory of locally finrec coalgebras. A roadmap of the major proof steps is visualized in Figure 3.

3.1 Applying F to Locally Finrec Coalgebras

At the heart of Lambek's lemma lies the observation that for every coalgebra $c\colon C \to FC$ application of F to the coalgebra structure yields the coalgebra $Fc\colon FC \to FFC$. This still holds for recursive coalgebras: If (R, r) is recursive, then so is (FR, Fr) (Corollary 2.19). However, for finrec coalgebras this does not hold, not even for very simple examples of set functors. For example, for the set functor $FX = \mathbb{N} \times X$, the coalgebra (FC, Fc) does not have a finite carrier unless C is empty.

Generalizing to locally finrec coalgebras, we recover the ability to apply F: if (C, c) is locally finrec, then so is (FC, Fc) provided that F preserves the colimit of a diagram of finrec coalgebra which forms (C, c). To stay in line with the Agda formalization, we allow $\mathscr{D}$ to be a large category.

Assumption 3.12 (🖐). We fix a locally finrec coalgebra (A, α) and a (potentially large) diagram $D\colon \mathscr{D} \to F$-Coalg of finrec coalgebras whose colimit is (A, α). We assume that $\mathscr{D}$ lies in Fil, and that F preserves the colimit of the diagram $V \circ D\colon \mathscr{D} \to \mathscr{C}$. We write (X_i, x_i) for the finrec coalgebra Di and $\pi_i\colon (X_i, x_i) \to (A, \alpha)$ for the colimit injection (in F-Coalg) for every $i \in \mathscr{D}$.

Remark 3.13. The names A and $\alpha\colon A \to FA$ are the members of the coalgebra record in the Agda's categories library. For our fixed locally finrec coalgebra we keep these names in order to stay as close as possible to the formalized proof, while all other coalgebras have roman letters as structures.

For an accessible functor F on a locally presentable category $\mathscr{C}$ (Examples 3.3 and 3.5), Assumption 3.12 is satisfied: the diagram D may be taken as the canonical projection

$$F\text{-Coalg}^{\mathsf{fin}}_{\mathsf{rec}}/(A, \alpha) \to F\text{-Coalg}.$$

In order to show that $(FA, F\alpha)$ is locally finrec again, we shall present a diagram $E\colon \mathscr{E} \to F$-Coalg of finrec coalgebras whose colimit is $(FA, F\alpha)$. For that we use that the object FA is the colimit of the canonical diagram $U_{FA}\colon \mathscr{C}_\mathsf{p}/FA \longrightarrow FA$. (Definition 3.1.(2c)). The objects of $\mathscr{C}_\mathsf{p}/FA$ are pairs (P, p) consisting of a presentable object $P \in \mathscr{C}_\mathsf{p}$ and a morphism $p\colon P \to FA$. The objects of the desired diagram scheme $\mathscr{E}$ are commuting triangles with one side in $\mathscr{C}_\mathsf{p}/FA$ and the other side coming from a colimit injection of D, which we denote by $\mathscr{E}_0$:

$$\mathscr{E}_0 := \{(P, p, i, p') \mid (P, p) \in \mathscr{C}_\mathsf{p}/FA,\ i \in \mathscr{D},\ F\pi_i \circ p' = p\} \quad (🖐)$$

$$(P, p, i, p') \in \mathscr{E}_0 \qquad \hat{=} \qquad \begin{array}{ccc} P & \xrightarrow{p} & FA \\ & {\scriptstyle p'}\searrow & \uparrow{\scriptstyle F\pi_i} \\ & & FX_i \end{array} \tag{5}$$

The elements of $\mathscr{E}_0$ relate the colimits of D and U_{FA}: the morphism p is the colimit injection of (P, p) to FA and π_i is the colimit injection of (X_i, x_i) to (A, α). The collection $\mathscr{E}_0$ is only as large as $\mathscr{D}$: if $\mathscr{D}$ is a small category, then $\mathscr{E}_0$ is a set.

Lemma 3.14. *Every object of $\mathscr{C}_\mathsf{p}/FA$ extends to an essentially unique object of $\mathscr{E}_0$:*

(1) *For every $(P, p) \in \mathscr{C}_\mathsf{p}/FA$ there exist $i \in \mathscr{D}$ and $p'\colon P \to FX_i$ such that $(P, p, i, p') \in \mathscr{E}_0$.* (🖐)

(2) *For all $(P, p, i, p') \in \mathscr{E}_0$ and $p''\colon P \to FX_i$ with $F\pi_i \circ p'' = p$, there exist $j \in \mathscr{D}$ and a $\mathscr{D}$-morphism $i \to j$ making p', p'' equal:*

$$P \underset{p''}{\overset{p'}{\rightrightarrows}} FX_i \xrightarrow{FD(i \to j)} FX_j. \quad (🖐)$$

Proof. Since F preserves the colimit of $V \circ D\colon \mathscr{D} \to \mathscr{C}$ we know that $F\pi_i\colon FX_i \to FA$ ($i \in \mathscr{D}$) is a colimit. Now apply Lemma 2.7 to this colimit and any given (P, p) in $\mathscr{C}_\mathsf{p}/FA$. □

Every object $t = (P, p, i, p') \in \mathscr{E}_0$ induces a finrec coalgebra: since P and X_i are in $\mathscr{C}_\mathsf{p}$, their coproduct $P + X_i$ exists (Assumption 3.7) and is presentable (Lemma 3.8). On $P + X_i$, we define the following coalgebra structure

$$\langle t \rangle := \left(P + X_i \xrightarrow{[p', x_i]} FX_i \xrightarrow{F\mathsf{inr}} F(P + X_i)\right)$$

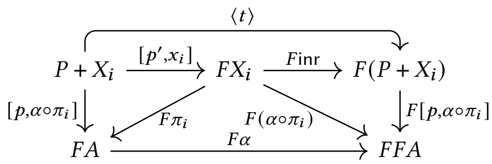
With Lemma 3.14.(1), this coalgebra structure can be understood to be generated by $p\colon P \to FA$. We put

$$E_0(t) := (P + X_i, \langle t \rangle) \in F\text{-Coalg}.$$

Moreover, we have a canonical coalgebra morphism

$$\mathsf{inj}_t := [p, \alpha \circ \pi_i] \colon E_0(t) \longrightarrow (FA, F\alpha); \qquad (✋)$$

indeed, consider the following diagram:

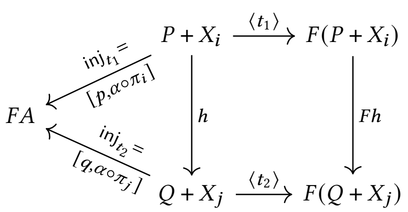


Its right-hand and middle parts commute due to functoriality of F, and for the left-hand part consider the coproduct components separately: the left-hand one commutes by (5) and the right-hand one since π_i is a coalgebra morphisms from (X_i, x_i) to (A, α).

In order to turn $\mathscr{E}_0$ into the category $\mathscr{E}$ and E_0 into an appropriate diagram $\mathscr{E} \to F$-Coalg, the morphisms need to be carefully chosen:

Definition 3.15 (✋). (1) The diagram scheme $\mathscr{E}$ is the full subcategory of F-Coalg$/(FA, F\alpha)$ given by the coalgebra morphisms $\mathsf{inj}_t \colon E_0 t \to (FA, F\alpha)$ $(t \in \mathscr{E}_0)$. In more detail:

- the objects of $\mathscr{E}$ are the elements of $\mathscr{E}_0$, and
- for $t_1, t_2 \in \mathscr{E}_0$, the hom set $\mathscr{E}(t_1, t_2)$ consists of those coalgebra morphisms $h \colon E_0(t_1) \to E_0(t_2)$ such that $\mathsf{inj}_{t_2} \circ h = \mathsf{inj}_{t_1}$:



for $t_1 = (P, p, i, p')$, $t_2 = (Q, q, j, q')$

(2) The diagram $E \colon \mathscr{E} \to F$-Coalg is defined by

$$E(P, p, i, p') = (P + X_i, \langle P, p, i, p' \rangle), \qquad E(h) = h.$$

Note that we do not need that $\mathscr{E}$ is filtered (or lies in Fil); by definition every colimit of finrec coalgebras is locally finrec.

Remark 3.16. (1) We obtain a canonical cocone on E given by

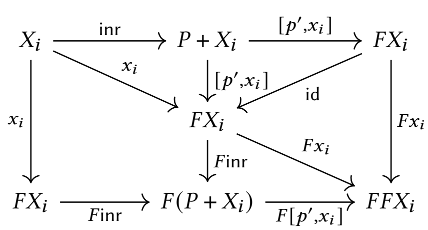
$$\mathsf{inj}_t \colon E(t) \to (FA, F\alpha) \qquad \text{for every } t \in \mathscr{E}_0.$$

(2) Note that it is important that $\mathscr{E}(t_1, t_2)$ contains *all* coalgebra morphisms of type $h \colon P_1 + X_{i_1} \to P_2 + X_{i_2}$ and not just coproducts $h_\ell + h_r \colon P_1 + X_{i_1} \to P_2 + X_{i_2}$. For in the latter case the colimit of E would simply be the coproduct $FA + A$.

All objects in the diagram E are indeed recursive:

Proposition 3.17 (✋). *For every object $t \in \mathscr{E}_0$, the coalgebra $E(t)$ is recursive.*

Proof. For $t = (P, p, i, p')$, consider the following diagram:



It clearly commutes so that we obtain two coalgebra morphisms:

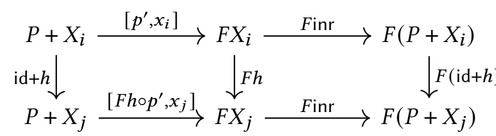
$$(X_i, x_i) \xrightarrow{\mathsf{inr}} (P + X_i, \langle t \rangle) \quad \text{and} \quad (P + X_i, \langle t \rangle) \xrightarrow{[p', x_i]} (FX_i, Fx_i)$$

By definition, $\langle t \rangle = F\mathsf{inr} \circ [p', x_i]$, so all requirements of Lemma 2.18 are met and thus $E(t) = (P + X_i, \langle t \rangle)$ is recursive. ◻

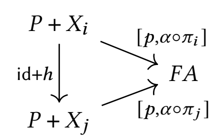
For the verification that $(FA, F\alpha)$ is locally finrec, it remains to show that the cocone $(\mathsf{inj}_t \colon VEt \to FA)_{t \in \mathscr{E}_0}$ is a colimit in $\mathscr{C}$ (Definition 3.9.(2b)). We do so by relating it to the colimits of $\mathscr{C}_\mathrm{p}/FA$ and $D \colon \mathscr{D} \to F$-Coalg. First a technical lemma stating that $\mathscr{D}$-morphism can be extended to $\mathscr{E}$-morphisms:

Lemma 3.18 (✋). *For every $\mathscr{D}$-morphism $f \colon i \to j$ we have an $\mathscr{E}$-morphism $\mathsf{id}_P + Df \colon (P, p, i, p') \to (P, p, j, FDf \circ p')$.*

Proof. Indeed, $h = Df \colon (X_i, x_i) \to (X_j, x_j)$ is a coalgebra morphism. Hence, the following diagram clearly commutes:

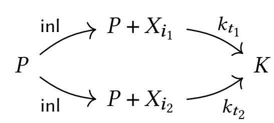


From $\pi_j \circ h = \pi_i$ we also obtain the commutative triangle

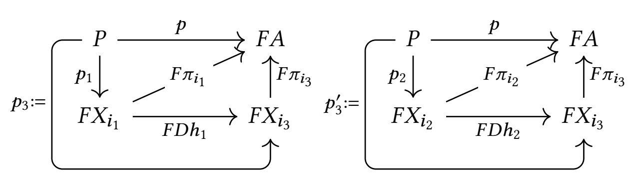


◻

Lemma 3.19 (✋). *For every $(V \circ E)$-cocone $(k_t \colon VEt \to K)_{t \in \mathscr{E}_0}$, the morphism k_t only depends on $(P, p) \in \mathscr{C}_\mathrm{p}/FA$. That is, for all objects $t_1 = (P, p, i_1, p_1)$ and $t_2 = (P, p, i_2, p_2)$ of $\mathscr{E}_0$ the following diagram commutes:*



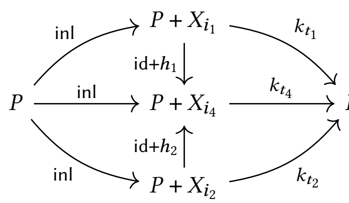
Proof. Using that $\mathscr{D}$ is filtered (Assumption 3.12), we first take an upper bound i_3 of i_1, i_2 in $\mathscr{D}$, that is, we have $\mathscr{D}$-morphisms $h_1 \colon i_1 \to i_3$ and $h_2 \colon i_2 \to i_3$. Using them we can extend i_3 to an object of $\mathscr{E}_0$ in two ways:



By the essential uniqueness (Lemma 3.14.(2)), there exist an $i_4 \in \mathscr{D}$ and a $\mathscr{D}$-morphism $m \colon i_3 \to i_4$ making p_3 and p'_3 equal:

$$p_4 := \Big(P \overset{p_3}{\underset{p'_3}{\rightrightarrows}} FX_{i_3} \xrightarrow{FDm} FX_{i_4} \Big)$$

Put $t_4 := (P, p, i_4, p_4)$ and use Lemma 3.18 to extend the $\mathscr{D}$-morphisms h_1 and h_2 to the $\mathscr{E}$-morphisms $\mathsf{id}_P + h_i \colon t_i \to t_4$, $i = 1, 2$. Using that the k_t form a cocone, we can then verify the desired independence property:



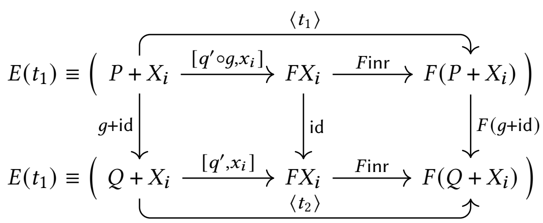
□

The independence following from Lemma 3.19 allows us to reduce cocones of $(V \circ E)$ to those of $\mathscr{C}_{\mathrm{p}}/FA$. For the latter we can then use the universal property of the colimit FA.

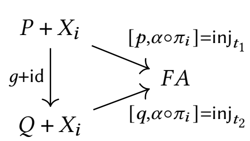
Lemma 3.20 (✍). *For every $(V \circ E)$-cocone $(k_t : VEt \to K)_{t \in \mathscr{E}_0}$, there is a U_{FA}-cocone $(\bar{k}_{(P,p)} : P \to K)_{(P,p)}$ such that $\bar{k}_{(P,p)} = k_t \circ \mathsf{inl}$ for some $t = (P, p, i, p') \in \mathscr{E}_0$.*

Proof. Given (P, p), we define $\bar{k}_{(P,p)} := k_t \circ \mathsf{inl}$ for the $t \in \mathscr{E}_0$ obtained from Lemma 3.14.(1). By Lemma 3.19, the morphism $\bar{k}_{(P,p)}$ is independent of the choice of i and p' in $t = (P, p, i, p')$.

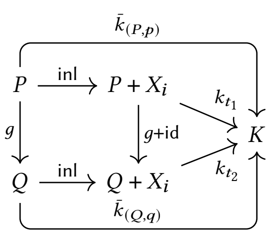
We now prove that every morphism $g : (P, p) \to (Q, q)$ in $\mathscr{C}_{\mathrm{p}}/FA$ can be extended to an $\mathscr{E}$-morphism as follows: for every extension $t_2 = (Q, q, i, q') \in \mathscr{E}_0$ of (Q, q) we have the morphism $g + \mathsf{id}_{X_i}$ in $\mathscr{E}$ from the extension $t_1 = (P, p, i, q' \circ g)$ of (P, p). Indeed, $g + \mathsf{id}_{X_i}$ is a coalgebra morphism from $E(t_1)$ to $E(t_2)$:



Moreover, using that $q \circ g = p$ we obtain $\mathsf{inj}_{t_2} \circ (g + \mathsf{id}_{X_i}) = \mathsf{inj}_{t_1}$:



The cocone coherence property of $(k_t)_{t \in \mathscr{E}_0}$ then yields that of the morphisms $\bar{k}_{(P,p)}$; indeed, the following diagram commutes:



□

For the verification of the universal property of the tentative colimit, we also translate the cocone morphisms back and forth:

Lemma 3.21. *For every $(V \circ E)$-cocone $(k_t : VEt \to K)_{t \in \mathscr{E}_0}$, a $\mathscr{C}$-morphism $v : FA \to K$ is a morphism of U_{FA}-cocones, that is,*

$$\bar{k}_{(P,p)} = \big(P \xrightarrow{p} FA \xrightarrow{v} K\big), \qquad \textit{for every } p \in \mathscr{C}_{\mathrm{p}}/FA,$$

if and only if it is a morphism of $(V \circ E)$-cocones:

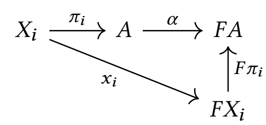
$$k_t = \big(P + X_i \xrightarrow{\mathsf{inj}_t} FA \xrightarrow{v} K\big), \qquad \textit{for every } t \in \mathscr{E}.$$

Proof. The implication for 'if' is easy to verify (✍): we have

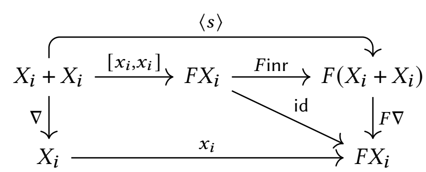
$$v \circ p = v \circ \underbrace{[p, \alpha \circ \pi_i]}_{\mathsf{inj}_t} \circ \mathsf{inl} = k_t \circ \mathsf{inl} = \bar{k}_{(P,p)},$$

using the definition of inj_t and $\bar{k}_{(P,p)}$ in the second and third steps, respectively.

The argument for 'only if' is non-trivial (✍). First recall from the proof of Proposition 3.17 that $\mathsf{inr} : (X_i, x_i) \to (P + X_i, \langle t \rangle)$ is a coalgebra morphism. Next we use that the object $\alpha \circ \pi_i : X_i \to FA$ of $\mathscr{C}_{\mathrm{p}}/FA$ has the following factorization using that $\pi_i : (X_i, x_i) \to (A, \alpha)$ is a coalgebra morphism:

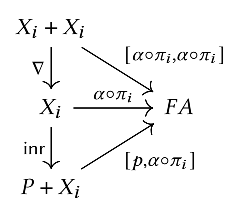


So we have the object $s = (X_i, \alpha \circ \pi_i, X_i, x_i)$ of $\mathscr{E}$ and we see that the codiagonal $\nabla : X_i + X_i \to X_i$ is a coalgebra morphism from $E(s)$ to (X_i, x_i):



(6)

Composing the coalgebra morphism $\mathsf{inr} : (X_i, x_i) \to (P + X_i, \langle t \rangle)$ with the one in (6) we obtain a morphism in $\mathscr{E}$ from s to t; indeed, the composition is a coalgebra morphism $E(s) \to E(t)$, and we have $\mathsf{inj}_t \circ \mathsf{inr} \circ \nabla = \mathsf{inj}_s$:



So we have an $\mathscr{E}$-morphism $\mathsf{inr} \circ \nabla : s \to t$ and conclude that

$$k_s = k_t \circ \mathsf{inr} \circ \nabla. \tag{7}$$

We are ready to show the desired equation $k_t = v \circ \mathsf{inj}_t$. We consider the coproduct components of the domain $P + X_i$ separately. For the left-hand component we have

$$\begin{aligned} v \circ \mathsf{inj}_t \circ \mathsf{inl} &= v \circ [p, \alpha \circ \pi_i] \circ \mathsf{inl} && \text{def. of } \mathsf{inj}_t \\ &= v \circ p && \text{since } [x, y] \circ \mathsf{inl} = x \\ &= \bar{k}_{(P,p)} && \text{by assumption} \\ &= k_t \circ \mathsf{inl} && \text{def. of } \bar{k}_{(P,p)} \text{ (Lemma 3.20).} \end{aligned}$$

For the right-hand coproduct component we compute as follows:

$$
\begin{aligned}
v \circ \mathsf{inj}_t \circ \mathsf{inr} = v \circ [p, \alpha \circ \pi_i] \circ \mathsf{inr} &= v \circ (\alpha \circ \pi_i) \\
&= \bar{k}_{(X_i, x_i)} && \text{by assump. for } p = \alpha \circ \pi_i \\
&= k_s \circ \mathsf{inl} && \text{def. of } \bar{k}_{(X_i,x_i)} \text{ (Lemma 3.20)} \\
&= k_t \circ \mathsf{inr} \circ \nabla \circ \mathsf{inl} && \text{by (7)} \\
&= k_t \circ \mathsf{inr} && \text{since } \nabla \circ \mathsf{inl} = \mathrm{id}.
\end{aligned}
$$

□

Finally, using that the canonical cocone

$$p \colon P \to FA \qquad ((P, p) \in \mathscr{C}_\mathsf{p}/FA)$$

is a colimit, we obtain that the $(V \circ E)$-cocone

$$\mathsf{inj}_t \colon VEt \to FA \qquad (t \in \mathscr{E})$$

is a colimit, too. Thus, under our Assumption 3.12 we have the following result:

Theorem 3.22 (). *The coalgebra $(FA, F\alpha)$ is locally finrec.*

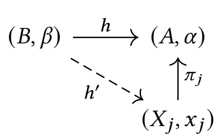
Proof. We shall prove that the cocone $\mathsf{inj}_t \colon P{+}X_i \to FA$ $(t \in \mathscr{E})$ is a colimit of $V \circ E$. Given a cocone $k_t \colon P{+}X_i \to K$ $(t \in \mathscr{E})$ we obtain the cocone $\bar{k}_{(P,p)} \colon P \to K$ $((P, p) \in \mathscr{C}_\mathsf{p}/FA)$ using Lemma 3.20. Thus, there exists a unique morphism $v \colon FA \to K$ such that $\bar{k}_{(P,p)} = v \circ p$ for every $p \colon P \to FA$ in $\mathscr{C}_\mathsf{p}/FA$. By Lemma 3.21, we have, equivalently, that $k_t = v \circ \mathsf{inj}_t$ for every $t \in \mathscr{E}$. Hence, v is the unique morphism with this property, which completes the proof. □

3.2 Unique Colimit Injections

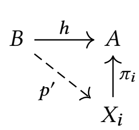
For the uniqueness condition in Lambek's lemma (Lemma 3.10.(2)) and the universal property in general, it helps to investigate when the colimit injections of locally finrec coalgebras are unique as coalgebra morphisms.

We continue to work under Assumption 3.12.

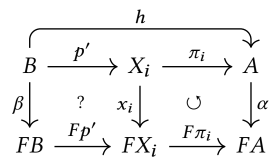
Lemma 3.23 (). *For every coalgebra (B, β) with presentable carrier B, every coalgebra morphism $h \colon (B, \beta) \to (A, \alpha)$ factorizes through one of the colimit injections $\pi_j \colon (X_j, x_j) \to (A, \alpha)$ in F-*Coalg*:*



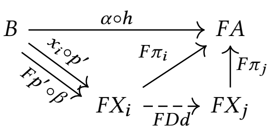
Proof. The hom-functor $\mathscr{C}(B, -) \colon \mathscr{C} \to \mathsf{Set}$ preserves the colimit A of $V \circ D$; here, we use that we have a colimit in the base category $\mathscr{C}$ (Definition 3.9.(2b)). Thus, we obtain an $i \in \mathscr{D}$ and a $\mathscr{C}$-morphism p' such that the following triangle commutes in $\mathscr{C}$ (cf. Lemma 2.7):



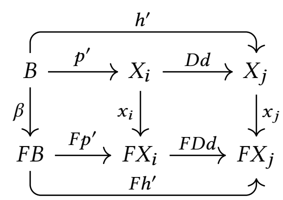
Proving that p' is a coalgebra morphism amounts to showing that the left-hand square of the following diagram commutes:



We will not prove its commutativity. Instead, observe that F preserves the colimit of $V \circ D$ (by assumption), so the morphisms $F\pi_i \colon FX_i \to FA$ $(i \in \mathscr{D})$ form a colimit. Since $\mathscr{D}$ lies in Fil, it is filtered. We now use that B is presentable and the ensuing essential uniqueness of factorizations of the morphism $\alpha \circ h \colon B \to FA$ through the colimit injection $F\pi_i$ (Lemma 2.7.(2)). The two paths of the left-hand square above are two such factorizations. Hence, there exists some morphism $d \colon i \to j$ in $\mathscr{D}$ such that $FDd \circ x_i \circ p' = FDd \circ Fp' \circ \beta$:



We verify that $h' := Dd \circ p' \colon B \to X_j$ is the desired coalgebra morphism $(B, \beta) \to (X_j, x_j)$:



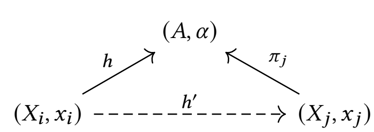
Indeed, the right-hand square commutes, and the left-hand one does when postcomposed with FDd; thus the outside commutes as desired. Moreover, we have

$$\pi_j \circ h' = \pi_j \circ Dd \circ p' = \pi_i \circ p' = h.$$

□

Theorem 3.24 (). *Given a locally finrec coalgebra (A, α) and $i \in \mathscr{D}$, the colimit injection π_i is the unique coalgebra morphism from (X_i, x_i) to (A, α) provided that $D \colon \mathscr{D} \to F$-*Coalg* is full.*

Proof. Given a coalgebra morphism $h \colon (X_i, x_i) \to (A, \alpha)$, apply Lemma 3.23 to obtain a factorization through some colimit injection:



Since D is full, $h' = Dd$ for some morphism $d \colon i \to j$ of $\mathscr{D}$. Hence, $h = \pi_j \circ Dd = \pi_i$ by the cocone coherence condition. □

3.3 Colimit of All Finrec Coalgebras

We now move on to consider the colimit of all finrec coalgebras and establish that this satisfies the two conditions in Lemma 3.10, which imply that it is the initial algebra.

Notation 3.25. We denote by $F\text{-}\mathsf{Coalg}^{\mathsf{fin}}_{\mathsf{rec}}$ the full subcategory of F-Coalg consisting of all finrec coalgebras with a carrier in $\mathscr{C}_\mathsf{p}$.

In the classical setting, this category is small: The reason is that $\mathscr{C}_\mathrm{p}$ is a set, and on each object of $\mathscr{C}_\mathrm{p}$ there is only a set of coalgebra structures. So the colimit of the forgetful functor

$$F\text{-Coalg}^{\mathsf{fin}}_{\mathsf{rec}} \xhookrightarrow{D} F\text{-Coalg} \xhookrightarrow{V} \mathscr{C}$$

exists whenever $\mathscr{C}$ is cocomplete.

Assumption 3.26 (✋). For the remainder of this section we assume that F-Coalg$^{\mathsf{fin}}_{\mathsf{rec}}$ lies in Fil, that the colimit of $V \circ D$ exists and is preserved by F.

We denote $\mathrm{colim}(V \circ D)$ by A and note that it carries a canonical coalgebra structure $\alpha\colon A \to FA$ such that $(A, \alpha) := \mathrm{colim}\, D$ (by Lemma 2.16). Moreover, the coalgebra (A, α) is locally finrec by definition.

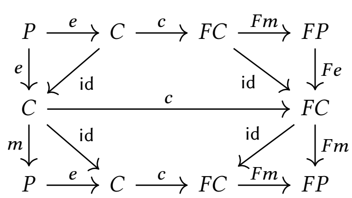
From the uniqueness result in Theorem 3.24 we deduce the following universal property:

Proposition 3.27 (✋). *For every finrec coalgebra (C, c), there is a unique coalgebra morphism $(C, c) \to (A, \alpha)$.*

Proof. By Lemma 3.6, every presentable object $C \in \mathscr{C}$ is a split quotient of some object P in $\mathscr{C}_\mathrm{p}$ via $e\colon P \twoheadrightarrow C$, say. Choose $m\colon C \rightarrowtail P$ such that $e \circ m = \mathrm{id}_C$. Then the following coalgebra structure

$$p = \big(P \xrightarrow{e} C \xrightarrow{c} FC \xrightarrow{Fm} FP\big)$$

turns e and m into a coalgebra morphism; indeed, the diagram below commutes



Thus, we have the coalgebra morphism $c \circ e\colon (P, p) \to (FC, Fc)$. By Lemma 2.18 applied to $h = m$ and $g = c \circ e$ (and noting that $p = Fh \circ g$) we see that (P, p) is recursive. Thus, this coalgebra lies in F-Coalg$^{\mathsf{fin}}_{\mathsf{rec}}$ since P lies in $\mathscr{C}_\mathrm{p}$. The diagram $D\colon F$-Coalg$^{\mathsf{fin}}_{\mathsf{rec}} \hookrightarrow F$-Coalg is a full functor. By Theorem 3.24, the colimit injection $\pi\colon (P, p) \to (A, \alpha)$ is the unique coalgebra morphism. Using that $e \circ m = \mathrm{id}_C$, we see that there is a unique coalgebra morphism from (C, c) to (A, α): we have the coalgebra morphism

$$(C, c) \xrightarrow{m} (P, p) \xrightarrow{\pi} (A, \alpha),$$

and given any coalgebra morphism $h\colon (C, c) \to (A, \alpha)$, we have $h \circ e = \pi$ by the unicity of π whence $h = h \circ e \circ m = \pi \circ m$. ◻

This universal property also lifts to colimits of finrec coalgebras:

Corollary 3.28 (✋). *For every locally finrec coalgebra (C, c), there is a unique coalgebra morphism $(C, c) \to (A, \alpha)$.*

Proof. Proposition 3.27 lifts from finrec coalgebras to locally finrec coalgebras by a general property (✋) of colimits. If $D\colon \mathscr{D} \to F$-Coalg is the witnessing diagram $(C, c) = \mathrm{colim}\, D$ that consists of finrec coalgebras Di $(i \in \mathscr{D})$, then there is a unique $Di \to (A, \alpha)$ for every $i \in \mathscr{D}$. Thus, (A, α) forms a cocone for D, which induces some morphism $(C, c) \to (A, \alpha)$. For uniqueness, consider $f, g\colon (C, c) \to (A, \alpha)$. For all $i \in \mathscr{D}$, we have $f \circ \mathsf{inj}_i = g \circ \mathsf{inj}_i\colon Di \to (A, \alpha)$, again by above uniqueness. Since colimit injections are jointly epic, this entails $f = g$. ◻

Corollary 3.29. *The coalgebra (A, α) is the terminal locally finrec coalgebra.*

This allows us to prove our main theorem.

Theorem 3.30 (✋). *The coalgebra structure $\alpha\colon A \to FA$ is an isomorphism, and $\alpha^{-1}\colon FA \to A$ is the initial F-algebra.*

Proof. Applying F to (A, α) yields a locally finrec coalgebra $(FA, F\alpha)$ (Theorem 3.22). By Corollary 3.28, we obtain a (unique) coalgebra morphism $(FA, F\alpha) \to (A, \alpha)$. By another application of Corollary 3.28, we see that identity is the only coalgebra morphism on (A, α).

Thus, α is an isomorphism by Lambek's lemma (Lemma 3.10), and by Lemma 2.20, its inverse is the structure of the initial F-algebra. ◻

Theorem 3.31. *For every accessible endofunctor on a locally presentable category, the initial algebra is the colimit of all recursive coalgebras with a λ-presentable carrier.*

Proof. Suppose that $\mathscr{C}$ is locally λ-presentable and that $F\colon \mathscr{C} \to \mathscr{C}$ is λ-accessible. Let Fil be the class of all λ-filtered categories. Then F-Coalg$^{\mathsf{fin}}_{\mathsf{rec}}$ is an essentially small category consisting of all recursive coalgebra with a λ-presentable carrier and the colimit of

$$F\text{-Coalg}^{\mathsf{fin}}_{\mathsf{rec}} \hookrightarrow F\text{-Coalg} \xrightarrow{V} \mathscr{C}$$

exists and is preserved by F. Thus, F has an initial algebra given by the colimit of the above diagram. ◻

4 COMPARISON WITH THE INITIAL-ALGEBRA CHAIN

The initial-algebra chain [5], which we have recalled for sets in the introduction, generalizes Kleene's fixed point theorem: recall that the latter starts with the bottom element and then successively applies a function to it. This yields an ascending chain approaching the desired fixed point from below.

For the construction of the initial algebra for an endofunctor $F\colon \mathscr{C} \to \mathscr{C}$, one starts with the initial object $0 \in \mathscr{C}$. Its initiality induces a morphism $!\colon 0 \longrightarrow F0$. Applying the functor successively to this morphism yields the ω-chain

$$0 \xrightarrow{!} F0 \xrightarrow{F!} F^2 0 \xrightarrow{F^2!} \cdots \xrightarrow{F^{k-1}!} F^k 0 \xrightarrow{F^k!} F^{k+1} 0 \xrightarrow{F^{k+1}!} \cdots$$

Let us write W_i for $F^i 0$ and $w_{i,j}\colon W_i \to W_j$ for the connecting morphisms. We also denote the colimit of this ω-chain by W_ω (assuming that it exists in $\mathscr{C}$).

(1) By the universal property of this colimit there is a canonical morphism $W_\omega \to FW_\omega$. If the colimit is preserved by F (e.g. because F is finitary), then this morphism has an inverse which can be shown to be the structure of an initial F-algebra.

(2) If the colimit is not preserved by F, then the iteration continues; the next step is induced by the universal property of the colimit W_ω:

$$W_\omega \longrightarrow FW_\omega \longrightarrow FFW_\omega \longrightarrow \cdots$$

The chain can be continued in this vein by transfinite recursion. If the functor F is λ-*accessible* [7, Def. 2.16] for some regular cardinal

λ, then the transfinite chain terminates in λ steps; this means that $w_{\lambda,\lambda+1}\colon W_\lambda \to FW_\lambda$ is an isomorphism. In contrast, our construction does not use transfinite recursion and takes only one colimit, regardless of the size of λ.

When looking at this chain through a coalgebraic lens, one observes that the chain consists of recursive coalgebras:

- $!\colon 0 \to F0$ is trivially a recursive coalgebra by initiality.
- Applying F to this coalgebra yields recursive coalgebras $F^k!\colon F^k 0 \to F(F^k 0)$, $k \in \mathbb{N}$ (Corollary 2.19).
- Their colimit $W_\omega \to FW_\omega$ is again recursive (Lemma 2.17).

However, in general these recursive coalgebras are not contained in the diagram scheme $F\text{-Coalg}^{\text{fin}}_{\text{rec}}$ that we use in our construction in Section 3. In every category, the initial object is presentable, so $!\colon 0 \to F0$ is a finrec coalgebra and thus a split quotient of a finrec coalgebra in $F\text{-Coalg}^{\text{fin}}_{\text{rec}}$ by Lemma 3.6. However, already the second step $F0 \to FF0$ of the initial-algebra chain may leave the realm of finrec coalgebras, because $F0$ may not be presentable anymore. Even for simple set functors such as $FX = \mathbb{N}+X\times X$, the set $F\emptyset = \mathbb{N}$ is infinite.

Of course, $F0 \to FF0$ and, more generally, $F^k 0 \to FF^k 0$ for every $k \in \mathbb{N}$ are *locally* finrec; this follows from Theorem 3.22.

5 AGDA FORMALIZATION

The non-trivial details concerning the definition of the diagram $E\colon \mathscr{E} \to F\text{-Coalg}$ (Definition 3.15) have motivated us to formalize the entire construction in a machine-checked setting.

5.1 Technical Aspects

After an attempt with Coq, we ultimately chose Agda (version 2.6.4) because it has an (almost official) library for category theory [14] (version 0.2.0). The formalized proofs are spread across 29 files and more than 5000 lines of code in total. The entire source code and compilation instructions can be found on

https://git8.cs.fau.de/software/initial-algebras-unchained
(also archived on archive.softwareheritage.org)

in the supplementary material archive.

All files compile with the flags `--without-K --safe`. For one file (`Iterate.Colimit`), we additionally use `--lossy-unification` to adjust Agda's unification heuristic and substantially speed up compilation.[2] This does not compromise correctness.

5.2 Formalization Challenges

Agda's type system is organized in levels: if a structure (like a function or record) quantifies over all sets on level ℓ, then the quantifying structure lives on level (at least) $\ell+1$. This implies that if we consider coalgebras living on level ℓ, then the property of being recursive lives on level $\ell+1$ because it quantifies over all algebras on level ℓ. Consequently, it is unclear whether a colimit of our main diagram $F\text{-Coalg}^{\text{fin}}_{\text{rec}} \hookrightarrow F\text{-Coalg}$ exists, even when restricting to coalgebras over Set. However, assuming the law of excluded middle, we can bring $F\text{-Coalg}^{\text{fin}}_{\text{rec}}$ back down to level ℓ (🐓). In order to keep the law of excluded middle out of the main construction, we allow potentially large colimits in Fil-accessible categories and in the definition of locally finrec coalgebras.

[2] https://agda.readthedocs.io/en/latest/language/lossy-unification.html

Contrary to our original expectations, no issues regarding choice principles arose. In our proofs, we have used multiple times that if a hom-functor preserves a colimit, then morphisms into the colimit factorize through the diagram (see e.g. Section 2.1 and Lemma 3.14). For the sake of modelling quotients, categories in the Agda library are not enriched over plain sets but instead over setoids. The latter are sets with an explicit equivalence relation denoting element equality. Thus, when working with elements of a colimit, in lieu of equivalence classes, one uses concrete representatives of equivalence classes. Note that it was surprisingly tedious to prove that setoids forms a Fil-accessible category (🐓).

6 CONCLUSIONS AND FUTURE WORK

We have shown that for a suitable endofunctor F on a Fil-accessible category, the initial algebra is obtained as the colimit of all recursive coalgebras with a presentable carrier. This formalizes the intuition that the initial algebra for F is formed by all data objects of type F modulo behavioural equivalence, which means that data objects are identified if they can be related by a coalgebra homomorphism.

Despite the fact that our description looks rather non-constructive, given that there is no concrete starting point, our main theorem can be proven in the constructive setting of Agda.

We leave as an open problem how well lfp categories can be formalized in a constructive setting with proper quotient types in lieu of setoids. In addition, it would be interesting to see whether our construction can be adapted to well-founded coalgebras.

ACKNOWLEDGMENTS

We thank Sergey Goncharov for many helpful discussions on Agda and `agda-categories` throughout the entire formalization process. We also thank Lutz Schröder for interesting comments on the initial-algebra chain, Henning Urbat for inspiring discussions on fixed point theorems, and Nathanael Arkor for helpful comments on notions related to lfp categories. We have benefited a lot from helpful answers in the `#agda` irc channel on `libera.chat`.

7 INDEX OF FORMALIZED RESULTS

Below we list the Agda file containing the referenced result and (if applicable) mention a concrete identifier in this file. The respective HTML files can be found on

https://arxiv.org/src/2405.09504/anc/index.html

and are also directly linked below.

Definition 2.2.(1) ≙ `preserves-colimit` in `Colimit-Lemmas`.
Definition 2.2.(2) ≙ `filtered` in `Filtered`.
Lemma 2.7 ≙ `hom-filtered-colimit-characterization` in `Hom-Colimit-Choice`.
Definition 2.8 in `Canonical-Cocone` – The file defines the category, the forgetful functor, and the cocone (3)
Lemma 2.16 in `F-Coalgebra-Colimit` – We have various formulations, depending on whether one needs the entire colimit or wants to prove that a particular cocone in coalgebras is colimitting.
Lemma 2.17 ≙ `Limitting-Cocone-IsRecursive` in `Coalgebra.Recursive`.
Lemma 2.18 ≙ `sandwich-recursive` in `Coalgebra.Recursive`.

Corollary 2.19 ≙ `iterate-recursive` in `Coalgebra.Recursive`.
Lemma 2.20 ≙ `iso-recursive⇒initial` in `Coalgebra.Recursive`.
Definition 3.1 ≙ `Accessible` in `Accessible-Category`.
Example 3.4 ≙ `Setoids-Accessible` in `Setoids-Accessible`.
Lemma 3.6 ≙ `presentable-split-in-fin` in `Accessible-Category`.
Lemma 3.8 ≙ `presentable-coproduct` in `Presentable`.
Definition 3.9.(1) ≙ `FiniteRecursive` in `Iterate.Assumptions` – A record with a parameter (the coalgebra) and two members: 1. the carrier is presentable, 2. the coalgebra is recursive.
Definition 3.9.(2) in `CoalgColim` – `CoalgColim` describes a coalgebra that occurs as the colimit of coalgebras all satisfying a certain property. This property is then instantiated to `FiniteRecursive` in the main construction.
Lemma 3.10 ≙ `lambek` in `Lambek` – The proof is surprisingly short and readable.
Assumption 3.12 ≙ `ProofGlobals` in `Iterate.ProofGlobals` – We maintain a record of all running assumptions such that we can fill the namespace with one line (`open ProofGlobals`).
Definition $\mathscr{E}_0$ ≙ $\mathscr{E}_0$ in `Iterate.FiniteSubcoalgebra`.
Lemma 3.14.(1) ≙ `P-to-triangle` in `Iterate.FiniteSubcoalgebra`.
Lemma 3.14.(2) ≙ `CC.p'-unique` in `Iterate.FiniteSubcoalgebra`.
inj_t ≙ `hom-to-FA` in `Iterate.FiniteSubcoalgebra` – Also see `hom-to-FA-i`$_1$ and `hom-to-FA-i`$_2$
Definition 3.15 ≙ $\mathscr{E}$ in `Iterate.DiagramScheme`.
Proposition 3.17 ≙ `P+X-coalg-is-FiniteRecursive` in `Iterate.FiniteSubcoalgebra`.
Lemma 3.18 ≙ `coalg-hom-to-`$\mathscr{E}$`-hom` in `Iterate.DiagramScheme`.
Lemma 3.19 ≙ `cocone-is-triangle-independent` in `Iterate.Colimit`.
Lemma 3.20 ≙ `E-Cocone-to-D` in `Iterate.Colimit` – Here, D refers to the canonical diagram $U_{FA}\colon \mathscr{C}_p/FA \to \mathscr{C}$
'if'-direction of Lemma 3.21 ≙ `reflect-Cocone⇒` in `Iterate.Colimit`.
'only if'-direction of Lemma 3.21 ≙ `lift-Cocone⇒` in `Iterate.Colimit`.
Theorem 3.22 ≙ `iterate-CoalgColimit` in `Iterate` – The entire proof spreads over the following modules of `Iterate`: Assumptions, Colimit, DiagramScheme, FiniteSubcoalgebra, ProofGlobals
Lemma 3.23 ≙ `hom-to-coalg-colim-triangle` in `Unique-Proj`.
Theorem 3.24 ≙ `unique-proj` in `Unique-Proj` – The colimit injections are called *projections* in the Formalization because this is the terminology in the `agda-categories` library
Assumption 3.26 ≙ `TerminalRecursive` in `Construction` – The assumptions are turned into module parameters. Instead of the essentially small F-$\mathsf{Coalg}^{\mathsf{fin}}_{\mathsf{rec}}$, the Agda code considers the colimit of those recursive coalgebras whose carrier lies in $\mathscr{C}_p$.
Proposition 3.27 ≙ `universal-property` in `Construction` – The statements about the constructed recursive coalgebra in the diagram are proven in retract-coalgebra-* in Coalgebra.Recursive
Corollary 3.28 ≙ `universal-property-locally-finrec` in `Construction`.
Colimit property in the proof of Corollary 3.28 ≙ `colimit-unique-rep` in `Colimit-Lemmas`.
Theorem 3.30 ≙ `initial-algebra` in `Construction`.
Level of F-$\mathsf{Coalg}^{\mathsf{fin}}_{\mathsf{rec}}$ ≙ `IsRecursive-via-LEM` in `Classical-Case`.
Setoids are Fil**-accessible** ≙ `Setoids-Accessible` in `Setoids-Accessible`.

REFERENCES

[1] Jirí Adámek, Francis Borceux, Stephen Lack, and Jirí Rosický. 2002. A classification of accessible categories. *J. Pure Appl. Algebra* 175 (2002), 7–30.
[2] Jirí Adámek, Stefan Milius, and Lawrence S. Moss. 2020. On Well-Founded and Recursive Coalgebras. In *Foundations of Software Science and Computation Structures - 23rd International Conference, FOSSACS 2020, Held as Part of the European Joint Conferences on Theory and Practice of Software, ETAPS 2020, Dublin, Ireland, April 25-30, 2020, Proceedings (Lecture Notes in Computer Science, Vol. 12077)*, Jean Goubault-Larrecq and Barbara König (Eds.). Springer, 17–36. https://doi.org/10.1007/978-3-030-45231-5_2
[3] Jirí Adámek, Stefan Milius, and Lawrence S. Moss. 2021. Initial Algebras Without Iteration ((Co)algebraic pearls). In *9th Conference on Algebra and Coalgebra in Computer Science, CALCO 2021, August 31 to September 3, 2021, Salzburg, Austria*. 5:1–5:20. https://doi.org/10.4230/LIPICS.CALCO.2021.5
[4] Jirí Adámek, Stefan Milius, and Jiri Velebil. 2006. Iterative algebras at work. *Math. Struct. Comput. Sci.* 16, 6 (2006), 1085–1131. https://doi.org/10.1017/S0960129506005706
[5] Jiří Adámek. 1974. Free algebras and automata realizations in the language of categories. *Commentationes Mathematicae Universitatis Carolinae* 015, 4 (1974), 589–602. http://eudml.org/doc/16649
[6] Jiří Adámek, Horst Herrlich, and George E. Strecker. 2004. Abstract and Concrete Categories. The Joy of Cats.
[7] Jiří Adámek and Jiří Rosický. 1994. *Locally Presentable and Accessible Categories*. Cambridge University Press.
[8] Steve Awodey. 2010. *Category Theory*. OUP Oxford. http://books.google.de/books?id=-MCJ6x2lC7oC
[9] Venanzio Capretta, Tarmo Uustalu, and Varmo Vene. 2006. Recursive coalgebras from comonads. *Inf. Comput.* 204, 4 (2006), 437–468. https://doi.org/10.1016/j.ic.2005.08.005
[10] Claudia Centazzo. 2004. *Generalized algebraic models*. Ph. D. Dissertation. Département de Mathématique de la Faculté des Sciences de l'Université catholique de Louvain.
[11] Claudia Centazzo, Jirí Rosický, and Enrico Vitale. 2004. A characterization of locally D-presentable categories. *Cah. Topol. Géom. Différ. Catég.* 45, 2 (2004), 141–146.
[12] Adam Eppendahl. 1999. Coalgebra-to-Algebra Morphisms. In *Conference on Category Theory and Computer Science, CTCS 1999, Edinburgh, UK, December 10-12, 1999 (Electronic Notes in Theoretical Computer Science, Vol. 29)*, Martin Hofmann, Giuseppe Rosolini, and Dusko Pavlovic (Eds.). Elsevier, 42–49. https://doi.org/10.1016/S1571-0661(05)80305-6
[13] Fredrik Nordvall Forsberg, Chuangjie Xu, and Neil Ghani. 2020. Three equivalent ordinal notation systems in cubical Agda. In *Proceedings of the 9th ACM SIGPLAN International Conference on Certified Programs and Proofs, CPP 2020, New Orleans, LA, USA, January 20-21, 2020*. 172–185. https://doi.org/10.1145/3372885.3373835
[14] Jason Z. S. Hu and Jacques Carette. 2021. Formalizing Category Theory in Agda. In *Proceedings of the 10th ACM SIGPLAN International Conference on Certified Programs and Proofs* (Virtual, Denmark) *(CPP 2021)*. Association for Computing Machinery, New York, NY, USA, 327–342. https://doi.org/10.1145/3437992.3439922
[15] Jean-Baptiste Jeannin, Dexter Kozen, and Alexandra Silva. 2017. Well-founded coalgebras, revisited. *Math. Struct. Comput. Sci.* 27, 7 (2017), 1111–1131. https://doi.org/10.1017/S0960129515000481
[16] Joachim Lambek. 1968. A Fixpoint Theorem for complete Categories. *Mathematische Zeitschrift* 103 (1968), 151–161. http://eudml.org/doc/170906
[17] Stefan Milius. 2010. A Sound and Complete Calculus for Finite Stream Circuits. In *Proceedings of the 25th Annual IEEE Symposium on Logic in Computer Science, LICS 2010, 11-14 July 2010, Edinburgh, United Kingdom*. IEEE Computer Society, 421–430. https://doi.org/10.1109/LICS.2010.11
[18] Stefan Milius, Dirk Pattinson, and Thorsten Wißmann. 2016. A New Foundation for Finitary Corecursion – The Locally Finite Fixpoint and Its Properties. In *Proc. 19th International Conference on Foundations of Software Science and Computation Structures (FoSSaCS 2016) (LNCS, Vol. 9634)*, Bart Jacobs and Christof Löding (Eds.). Springer, 107–125. https://doi.org/10.1007/978-3-662-49630-5_7
[19] Stefan Milius, Dirk Pattinson, and Thorsten Wißmann. 2020. A new foundation for finitary corecursion and iterative algebras. *Inf. Comput.* 271 (2020), 104456.

https://doi.org/10.1016/j.ic.2019.104456

[20] Gerhard Osius. 1974. Categorical set theory: A characterization of the category of sets. *Journal of Pure and Applied Algebra* 4, 1 (1974), 79–119. https://doi.org/10.1016/0022-4049(74)90032-2

[21] Andrew M. Pitts and S. C. Steenkamp. 2021. Constructing Initial Algebras Using Inflationary Iteration. In *Proceedings of the Fourth International Conference on Applied Category Theory, ACT 2021, Cambridge, United Kingdom, 12-16th July 2021*. 88–102. https://doi.org/10.4204/EPTCS.372.7

[22] Tianyu Sun, Wensheng Yu, and Yaoshun Fu. 2019. Formalization of Transfinite Induction in Coq. In *2019 Chinese Automation Congress (CAC)*. 1001–1005. https://doi.org/10.1109/CAC48633.2019.8997376

[23] Paul Taylor. 1996. Intuitionistic Sets and Ordinals. *J. Symb. Log.* 61, 3 (1996), 705–744. https://doi.org/10.2307/2275781

[24] Paul Taylor. 1999. *Practical Foundations of Mathematics*. Cambridge studies in advanced mathematics, Vol. 59. Cambridge University Press.

[25] Paul Taylor. 2023. *Well Founded Coalgebras and Recursion*. https://www.paultaylor.eu/ordinals/welfcr.pdf accessed on 2024-01-27.

[26] Henning Urbat. 2017. Finite Behaviours and Finitary Corecursion. In *7th Conference on Algebra and Coalgebra in Computer Science, CALCO 2017, June 12-16, 2017, Ljubljana, Slovenia*. 24:1–24:16. https://doi.org/10.4230/LIPIcs.CALCO.2017.24

[27] Ernst Zermelo. 1904. Beweis, daß jede Menge wohlgeordnet werden kann. *Math. Ann.* 59 (1904), 514–516.