

# A New Foundation for Finitary Corecursion

## The Locally Finite Fixpoint and Its Properties

Stefan Milius<sup>1</sup>, Dirk Pattinson<sup>2</sup>, and Thorsten Wißmann<sup>1</sup>(✉)

<sup>1</sup> Friedrich-Alexander-Universität Erlangen-Nürnberg, Erlangen, Germany  
thorsten.wissmann@fau.de

<sup>2</sup> The Australian National University, Canberra, Australia

**Abstract.** This paper contributes to a theory of the behaviour of “finite-state” systems that is generic in the system type. We propose that such systems are modeled as coalgebras with a finitely generated carrier for an endofunctor on a locally finitely presentable category. Their behaviour gives rise to a new fixpoint of the coalgebraic type functor called *locally finite fixpoint* (LFF). We prove that if the given endofunctor preserves monomorphisms then the LFF always exists and is a subcoalgebra of the final coalgebra (unlike the rational fixpoint previously studied by Adámek, Milius and Velebil). Moreover, we show that the LFF is characterized by two universal properties: 1. as the final locally finitely generated coalgebra, and 2. as the initial fg-iterative algebra. As instances of the LFF we first obtain the known instances of the rational fixpoint, e.g. regular languages, rational streams and formal power-series, regular trees etc. And we obtain a number of new examples, e.g. (realtime deterministic resp. non-deterministic) context-free languages, constructively  $S$ -algebraic formal power-series (and any other instance of the generalized powerset construction by Silva, Bonchi, Bonsangue, and Rutten) and the monad of Courcelle’s algebraic trees.

## 1 Introduction

Coalgebras capture many types of state based system within a uniform and mathematically rich framework [39]. One outstanding feature of the general theory is *final semantics* which gives a fully abstract account of system behaviour. For example, coalgebraic modelling of deterministic automata (without a finiteness restriction on state sets) yields the set of all formal languages as a final model, and restricting to *finite* automata one precisely obtains the regular languages [38]. This correspondence has been generalized to locally finitely presentable categories [8, 20], where *finitely presentable* objects play the role of finite sets, leading to the notion of *rational fixpoint* that provides final semantics to all models with finitely presentable carrier [30]. It is known that the rational fixpoint is fully abstract (identifies all behaviourally equivalent states) as long as finitely presentable objects agree with finitely generated objects in the base category

---

S. Milius and T. Wißmann—Supported by Deutsche Forschungsgemeinschaft (DFG) under project MI 717/5-1.

[12, Proposition 3.12]. While this is the case in some categories (e.g. sets, posets, graphs, vector spaces, commutative monoids), it is currently unknown in other base categories that are used in the construction of system models, for example in idempotent semirings (used in the treatment of context-free grammars [43]), in algebras for the stack monad (used for modelling configurations of stack machines [23]); or it even fails, for example in the category of finitary monads on sets (used in the categorical study of algebraic trees [7]), or Eilenberg-Moore categories for a monad in general (the target category of generalized determinization [41], in which the above examples live). Coalgebras over a category of Eilenberg-Moore algebras over  $\mathbf{Set}$  in particular provide a paradigmatic setting: automata that describe languages beyond the regular languages consist of a finite state set, but their transitions produce side effects such as the manipulation of a stack. These can be described by a monad, so that the (infinite) set of system states (machine states plus stack content) is described by a free algebra (for that monad) that is generated by the finite set of machine states. This is formalized by the generalized powerset construction [41] and interacts nicely with the coalgebraic framework we present.

Technically, the shortcoming of the rational fixpoint is due to the fact that finitely presentable objects are not closed under quotients, so that the rational fixpoint itself may fail to be a subcoalgebra of the final coalgebra and so identifies too little behaviour. The main conceptual contribution of this paper is the insight that also in cases where finitely presentable and finitely generated do not agree, the *locally finite fixpoint* provides a fully abstract model of finitely generated behaviour. We give a construction of the locally finite fixpoint, and support our claim both by general results and concrete examples: we show that under mild assumptions, the locally finite fixpoint always exists, and is indeed a subcoalgebra of the final coalgebra. Moreover, we give a characterization of the locally finite fixpoint as the initial iterative algebra. We then instantiate our results to several scenarios studied in the literature.

First, we show that the locally finite fixpoint is universal (and fully abstract) for the class of systems produced by the generalized powerset construction over  $\mathbf{Set}$ : every determinized finite-state system induces a unique homomorphism to the locally finite fixpoint, and the latter contains precisely the finite-state behaviours.

Applied to the coalgebraic treatment of context-free languages, we show that the locally finite fixpoint yields precisely the context-free languages, and real-time deterministic context-free languages, respectively, when modelled using algebras for the stack monad of [23]. For context-free languages weighted in a semiring  $S$ , or equivalently for constructively  $S$ -algebraic power series [36], the locally finite fixpoint comprises precisely those, by phrasing the results of Winter et al. [44] in terms of the generalized powerset construction. Our last example shows the applicability of our results beyond categories of Eilenberg-Moore algebras over  $\mathbf{Set}$ , and we characterize the monad of Courcelle's algebraic trees over a signature [7, 16] as the locally finite fixpoint of an associated functor (on a category of monads), solving an open problem of [7].

The work presented here is based on the third author's master thesis in [45]. Most proofs are omitted; they can be found in the full version [33] of our paper.

## 2 Preliminaries and Notation

**Locally Finitely Presentable Categories.** A *filtered colimit* is the colimit of a diagram  $\mathcal{D} \rightarrow \mathcal{C}$  where  $\mathcal{D}$  is filtered (every finite subdiagram has a cocone in  $\mathcal{D}$ ) and *directed* if  $\mathcal{D}$  is additionally a poset. *Finitary functors* preserve filtered (equivalently directed) colimits. Objects  $C \in \mathcal{C}$  are *finitely presentable* (fp) if the hom-functor  $\mathcal{C}(C, -)$  preserves filtered (equivalently directed) colimits, and *finitely generated* (fg) if  $\mathcal{C}(C, -)$  preserves directed colimits of monos (i.e. colimits of directed diagrams where all connecting morphisms are monic). Clearly any fp object is fg, but not vice versa. Also, fg objects are closed under strong epis (quotients) which fails for fp objects in general. A cocomplete category is *locally finitely presentable* (lfp) if the full subcategory  $\mathcal{C}_{\text{fp}}$  of finitely presentable objects is essentially small, i.e. is up to isomorphism only a set, and every object  $C \in \mathcal{C}$  is a filtered colimit of a diagram in  $\mathcal{C}_{\text{fp}}$ . We refer to [8, 20] for further details.

It is well known that the categories of sets, posets and graphs are lfp with finitely presentable objects precisely the finite sets, posets, graphs, respectively. The category of vector spaces is lfp with finite-dimensional spaces being fp. Every finitary variety is lfp (i.e. an equational class of algebras induced by finiteness operations or equivalently the Eilenberg-Moore category for a finitary  $\text{Set-Monad}$ , see Sect. 4.1 later). The finitely generated objects are the finitely generated algebras, and finitely presentable objects are algebras specified by finitely many generators and relations. This includes the categories of groups, monoids, (idempotent) semirings, semi-modules, etc. Every lfp category has mono/strong epi factorization [8, Proposition 1.16], i.e. every  $f$  factors as  $f = m \cdot e$  with  $m$  mono (denoted by  $\mapsto$ ),  $e$  strong epi (denoted by  $\twoheadrightarrow$ ), and we call the domain  $\text{Im}(f)$  of  $e$  the *image* of  $f$ . Any strong epi  $e$  has the diagonal fill-in property, i.e.  $m \cdot g = h \cdot e$  with  $m$  mono and  $e$  strong epi gives a unique  $d$  such that  $m \cdot d = h$  and  $g = d \cdot e$ .

**Coalgebras.** If  $H : \mathcal{C} \rightarrow \mathcal{C}$  is an endofunctor,  $H$ -*coalgebras* are pairs  $(C, c)$  with  $c : C \rightarrow HC$ , and  $C$  is the *carrier* of  $(C, c)$ . Homomorphisms  $f : (C, c) \rightarrow (D, d)$  are maps  $f : C \rightarrow D$  such that  $Hf \cdot c = d \cdot f$ . This gives a category denoted by  $\text{Coalg}H$ . If its final object exists then this final  $H$ -coalgebra  $(\nu H, \tau)$  represents a canonical domain of behaviours of  $H$ -typed systems, and induces for each  $(C, c)$  a unique homomorphism, denoted by  $c^\dagger$ , giving semantics to the system  $(C, c)$ . The final coalgebra always exists provided  $\mathcal{C}$  is lfp and  $H$  is finitary. The forgetful functor  $\text{Coalg}H \rightarrow \mathcal{C}$  creates colimits and reflects monos and epis. A morphism  $f$  in  $\text{Coalg}H$  is *mono-carried* (resp. *epi-carried*) if the underlying morphism in  $\mathcal{C}$  is monic (resp. epic). Strong epi/mono factorizations lift from  $\mathcal{C}$  to  $\text{Coalg}H$  whenever  $H$  preserves monos yielding epi-carried/mono-carried factorizations. A *directed union of coalgebras* is the colimit of a directed diagram in  $\text{Coalg}H$  where all connecting morphisms are mono-carried.

**The Rational Fixpoint.** For  $\mathcal{C}$  lfp and  $H : \mathcal{C} \rightarrow \mathcal{C}$  finitary let  $\text{Coalg}_{\text{fp}}H$  denote the full subcategory of  $\text{Coalg}H$  of coalgebras with fp carrier, and  $\text{Coalg}_{\text{lfp}}H$  the full subcategory of  $\text{Coalg}H$  of coalgebras that arise as filtered colimits of coalgebras with fp carrier [30, Corollary III.13]. The coalgebras in  $\text{Coalg}_{\text{lfp}}H$  are

called *lfp coalgebras* and for  $\mathcal{C} = \mathbf{Set}$  those are precisely the locally finite coalgebras (i.e. those coalgebras where every element is contained in a finite subcoalgebra). The final lfp coalgebra exists and is the colimit of the inclusion  $\mathbf{Coalg}_{\text{fp}}H \hookrightarrow \mathbf{Coalg}H$ , and it is a fixpoint of  $H$  (see [6]) called the *rational fixpoint* of  $H$ . Here are some examples: the rational fixpoint of a polynomial set functor associated to a finitary signature  $\Sigma$  is the set of rational  $\Sigma$ -trees [6], i.e. finite and infinite  $\Sigma$ -trees having, up to isomorphism, finitely many subtrees only, and one obtains rational weighted languages for Noetherian semirings  $S$  for a functor on the category of  $S$ -modules [12], and rational  $\lambda$ -trees for a functor on the category of presheaves on finite sets [2] or for a related functor on nominal sets [34]. If the classes of fp and fg objects coincide in  $\mathcal{C}$ , then the rational fixpoint is a subcoalgebra of the final coalgebra [12, Theorem 3.12]. This is the case in the above examples, but not in general, see [12, Example 3.15] for a concrete example where the rational fixpoint does not identify behaviourally equivalent states. Conversely, even if the classes differ, the rational fixpoint can be a subcoalgebra, e.g. for any constant functor.

**Iterative Algebras.** If  $H : \mathcal{C} \rightarrow \mathcal{C}$  is an endofunctor, an  $H$ -algebra  $(A, a : HA \rightarrow A)$  is *iterative* if every *flat equation morphism*  $e : X \rightarrow HX + A$  where  $X$  is an fp object has a unique *solution*, i.e. if there exists a unique  $e^\dagger : X \rightarrow A$  such that  $e^\dagger = [a, \text{id}_A] \cdot (He^\dagger + \text{id}_A) \cdot e$ . The rational fixpoint is also characterized as the initial iterative algebra [6] and is the starting point of the coalgebraic approach to Elgot’s iterative theories [18] and to the iteration theories of Bloom and Ésik [3, 4, 6, 11].

### 3 The Locally Finite Fixpoint

The locally finite fixpoint can be characterized similarly to the rational fixpoint, but with respect to coalgebras with finitely generated (not finitely presentable) carrier. We show that the locally finite fixpoint always exists, and is a subcoalgebra of the final coalgebra, i.e. identifies all behaviourally equivalent states. As a consequence, the locally finite fixpoint provides a fully abstract notion of finitely generated behaviour. From now on, we rely on the following:

**Assumption 3.1.** *Throughout the rest of the paper we assume that  $\mathcal{C}$  is an lfp category and that  $H : \mathcal{C} \rightarrow \mathcal{C}$  is finitary and preserves monomorphisms.*

As for the rational fixpoint, we denote the full subcategory of  $\mathbf{Coalg}H$  comprising all coalgebras with finitely generated carrier by  $\mathbf{Coalg}_{\text{fg}}H$  and have the following notion of locally finitely generated coalgebra.

**Definition 3.2.** *A coalgebra  $X \xrightarrow{x} HX$  is called locally finitely generated (lfg) if for all  $f : S \rightarrow X$  with  $S$  finitely generated, there exist a coalgebra  $p : P \rightarrow HP$  in  $\mathbf{Coalg}_{\text{fg}}H$ , a coalgebra morphism  $h : (P, p) \rightarrow (X, x)$  and some  $f' : S \rightarrow P$  such that  $h \cdot f' = f$ .  $\mathbf{Coalg}_{\text{lfg}}H \subseteq \mathbf{Coalg}H$  denotes the full subcategory of lfg coalgebras.*

Equivalently, one can characterize lfg coalgebras in terms of subobjects and subcoalgebras, making it a generalization of *local finiteness* in **Set**, i.e. the property of a coalgebra that every element is contained in a finite subcoalgebra.

**Lemma 3.3.**  $X \xrightarrow{x} HX$  is an lfg coalgebra iff for all fg subobjects  $S \xrightarrow{f} X$ , there exist a subcoalgebra  $h : (P, p) \rightarrow (X, x)$  and a mono  $f' : S \rightarrow P$  with  $h \cdot f' = f$ , i.e.  $S$  is a subobject of  $P$ .

*Proof.* ( $\Rightarrow$ ) Given some mono  $f : S \rightarrow X$ , factor the induced  $h$  into some strong epi-carried and mono-carried homomorphisms and use that fg objects are closed under strong epis. ( $\Leftarrow$ ) Factor  $f : S \rightarrow X$  into an epi and a mono  $g : \text{Im}(f) \rightarrow X$  and use the diagonal fill-in property for  $g$ .  $\square$

Evidently all coalgebras with finitely generated carriers are lfg. Moreover, lfg coalgebras are precisely the filtered colimits of coalgebras from  $\text{Coalg}_{\text{fg}}H$ .

**Proposition 3.4.** Every filtered colimit of coalgebras from  $\text{Coalg}_{\text{fg}}H$  is lfg.

*Proof (Sketch).* One first proves that directed unions of coalgebras from  $\text{Coalg}_{\text{fg}}H$  are lfg. Now given a filtered colimit  $c_i : X_i \rightarrow C$  where  $X_i$  are coalgebras in  $\text{Coalg}_{\text{fg}}H$ , one epi-mono factorizes every colimit injection:  $c_i = (X_i \xrightarrow{e_i} T_i \xrightarrow{m_i} C)$ . Using the diagonalization of the factorization one sees that the  $T_i$  form a directed diagram of subobjects of  $C$ . Furthermore  $C$  is the directed union of the  $T_i$  and therefore an lfg coalgebra as desired.  $\square$

**Proposition 3.5.** Every lfg coalgebra  $(X, x)$  is a directed colimit of its subcoalgebras from  $\text{Coalg}_{\text{fg}}H$ .

*Proof.* Recall from [8, Proof of Theorem 1.70] that  $X$  is the colimit of the diagram of all its finitely generated subobjects. Now the subdiagram given by all subcoalgebras of  $X$  is cofinal. Indeed, this follows directly from the fact that  $(X, x)$  is an lfg coalgebra: for every subobject  $S \rightarrow X$ ,  $S$  fg, we have a subcoalgebra of  $(X, x)$  in  $\text{Coalg}_{\text{fg}}H$  containing  $S$ .  $\square$

**Corollary 3.6.** The lfg coalgebras are precisely the filtered colimits, or equivalently directed unions, of coalgebras with fg carrier.

As a consequence, a coalgebra is final in  $\text{Coalg}_{\text{fg}}F$  if there is a unique morphism from every coalgebra with finitely generated carrier.

**Proposition 3.7.** An lfg coalgebra  $L$  is final in  $\text{Coalg}_{\text{fg}}H$  iff for every coalgebra  $X$  in  $\text{Coalg}_{\text{fg}}H$  there exists a unique coalgebra morphism from  $X$  to  $L$ .

The proof is analogous to [30, Theorem 3.14]; the full argument can be found in [33]. Cocompleteness of  $\mathcal{C}$  ensures that the final lfg coalgebra always exists.

**Theorem 3.8.** The category  $\text{Coalg}_{\text{fg}}H$  has a final object, and the final lfg coalgebra is the colimit of the inclusion  $\text{Coalg}_{\text{fg}}H \hookrightarrow \text{Coalg}_{\text{lfg}}H$ .

*Proof.* By Corollary 3.6, the colimit of the inclusion  $\text{Coalg}_{\text{fg}}H \hookrightarrow \text{Coalg}_{\text{lfg}}H$  is the same as the colimit of the entire  $\text{Coalg}_{\text{lfg}}H$ . And the latter is clearly the final object of  $\text{Coalg}_{\text{lfg}}H$ .  $\square$

This theorem provides a construction of the final lfg coalgebra collecting precisely the behaviours of the coalgebras with fg carriers. In the following we shall show that this construction does indeed identify precisely behaviourally equivalent states, i.e. the final lfg coalgebra is always a subcoalgebra of the final coalgebra. Just like fg objects are closed under quotients – in contrast to fp objects – we have a similar property of lfg coalgebras:

**Lemma 3.9.** *Lfg coalgebras are closed under strong quotients, i.e. for every strong epi carried coalgebra homomorphisms  $X \twoheadrightarrow Y$ , if  $X$  is lfg then so is  $Y$ .*

The failure of this property for lfp coalgebras is the reason why the rational fixpoint is not necessarily a subcoalgebra of the final coalgebra and in particular the rational fixpoint in [12, Example 3.15] is an lfp coalgebra for which the property fails.

**Theorem 3.10.** *The final lfg  $H$ -coalgebra is a subcoalgebra of the final  $H$ -coalgebra.*

*Proof.* Let  $(L, \ell)$  be the final lfg coalgebra. Consider the unique coalgebra morphism  $L \rightarrow \nu H$  and take its factorization:

$$\text{id} \begin{array}{c} \dashrightarrow \\ \curvearrowright \\ \dashrightarrow \end{array} (L, \ell) \begin{array}{c} \xrightarrow{e} \\ \dashrightarrow \\ \xleftarrow{i^\dagger} \end{array} (I, i) \xrightarrow{m} (\nu H, \tau) \text{ , with } e \text{ strong epi in } \mathcal{C}.$$

By Lemma 3.9,  $I$  is an lfg coalgebra and so by finality of  $L$  we have the coalgebra morphism  $i^\dagger$  such that  $\text{id}_L = i^\dagger \cdot e$ . It follows that  $e$  is monic and therefore an iso.  $\square$

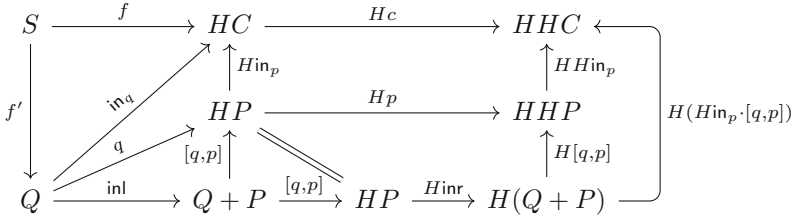
In other words, the final lfg  $H$ -coalgebra collects precisely the finitely generated behaviours from the final  $H$ -coalgebra. We now show that the final lfg coalgebra is a fixpoint of  $H$  which hinges on the following:

**Lemma 3.11.** *For any lfg coalgebra  $C \xrightarrow{c} HC$ , the coalgebra  $HC \xrightarrow{Hc} HHC$  is lfg.*

*Proof.* Consider  $f : S \rightarrow HC$  with  $S$  finitely generated. As  $\mathcal{C}$  is lfp we know that  $HC$  is the colimit of its fg subobjects, and so  $f : S \rightarrow HC$  factors through some subobject  $\text{in}_q : Q \twoheadrightarrow HC$  with  $Q$  fg and  $f = \text{in}_q \cdot f'$ . On the other hand,  $(C, c)$  is lfg, i.e. the directed union of its subcoalgebras from  $\text{Coalg}_{\text{fg}}H$ . Then, since  $H$  is finitary and mono-preserving,  $HC \xrightarrow{c} HHC$  is also a directed union and the morphism  $\text{in}_q : Q \rightarrow HC$  factors through some  $HP \xrightarrow{Hp} HHP$  with  $(P, p) \in \text{Coalg}_{\text{fg}}H$  via  $\text{in}_p : (P, p) \twoheadrightarrow (C, c)$ , i.e.  $H\text{in}_p \cdot q = \text{in}_q$ . Finally, we can construct a coalgebra with fg carrier

$$Q + P \xrightarrow{[q,p]} HP \xrightarrow{H\text{in}_p} H(Q + P)$$

and a coalgebra homomorphism  $H\text{in}_p \cdot [q, p] : Q + P \rightarrow HC$ . In the diagram



every part trivially commutes, so  $Hin_p \cdot [q, p]$  is the desired homomorphism.  $\square$

So with a proof in virtue to Lambek’s Lemma [28, Lemma 2.2], we obtain the desired fixpoint:

**Theorem 3.12.** *The carrier of the final lfg  $H$ -coalgebra is a fixpoint of  $H$ .*

We denote the above fixpoint by  $(\vartheta H, \ell)$  and call it the *locally finite fixpoint* (LFF) of  $H$ . In particular, the LFF always exists under Assumption 3.1, providing a finitary corecursion principle.

*Remark 3.13.* As we mentioned in the introduction the rational fixpoint of the finitary functor  $H$  is the initial iterative algebra for  $H$ . A similar algebraic characterization is possible for the LFF. One simply replaces the fp object  $X$  in the definition of a flat equation morphism by an fg object to obtain the notion of an fg-iterative algebra.

**Theorem 3.14.** *The LFF is the initial fg-iterative  $H$ -algebra.*

For details, see the full version [33] of our paper or [45].

**Relation to the Rational Fixpoint.** There are examples, where the rational fixpoint is not a subcoalgebra of the final coalgebra. In categories, where fp and fg objects coincide, the rational fixpoint and the LFF coincide as well (cf. the respective colimit-construction in Sect. 2 and Theorem 3.8). In this section we will see, under slightly stronger assumptions, that fg-carried coalgebras are quotients of fp-carried coalgebras, and in particular the locally finite fixpoint is a quotient of the rational fixpoint: namely its image in the final coalgebra.

**Assumption 3.15.** *In addition to Assumption 3.1, assume that in the base category  $\mathcal{C}$ , every finitely presentable object is a strong quotient of a finitely presentable strong epi projective object and that the endofunctor  $H$  also preserves strong epis.*

The condition that every fg object is the strong quotient of a strong epi projective often is phrased as *having enough strong epi projectives* [14]. This assumption is apparently very strong but still is met in many situations:

*Example 3.16*

- In categories in which all (strong) epis are split, every object is projective and any endofunctor preserves epis, e.g. in  $\mathbf{Set}$  or  $\mathbf{Vec}_K$ .

- In the category of finitary endofunctors  $\text{Fun}_f(\text{Set})$ , all polynomial functors are projective. The finitely presentable functors are quotients of polynomial functors  $H_\Sigma$ , where  $\Sigma$  is a finite signature.
- In the Eilenberg-Moore category  $\text{Set}^T$  for a finitary monad  $T$ , strong epis are surjective  $T$ -algebra homomorphisms, and thus preserved by any endofunctor. In  $\text{Set}^T$ , every free algebra  $TX$  is projective; this is easy to see using the projectivity of  $X$  in  $\text{Set}$ . Every finitely generated object of  $\text{Set}^T$  is a strong quotient of some free algebra  $TX$  with  $X$  finite. For more precise definitions, see Sect. 4.1 later.

**Proposition 3.17.** *Every coalgebra in  $\text{Coalg}_{\text{fg}}H$  is a strong quotient of a coalgebra with finitely presentable carrier.*

**Theorem 3.18.**  *$\vartheta H$  is the image of the rational fixpoint  $\varrho H$  in the final coalgebra.*

*Proof.* Consider the factorization  $(\varrho H, r) \xrightarrow{e} (B, b) \xrightarrow{m} (\nu H, \tau)$ . Since  $\varrho H$  is the colimit of all fp carried  $H$ -coalgebras it is an lfg coalgebra by Proposition 3.4 using that fp objects are also fg. Hence, by Lemma 3.9 the coalgebra  $B$  is lfg, too. By Proposition 3.7 it now suffices to show that from every  $(X, x) \in \text{Coalg}_{\text{fg}}H$  there exists a unique coalgebra morphism into  $(B, b)$ . Given  $(X, x)$  in  $\text{Coalg}_{\text{fg}}H$ , it is the quotient  $q : (P, p) \rightarrow (X, x)$  of an fp-carried coalgebra by Proposition 3.17. Hence, we obtain a unique coalgebra morphism  $p^\dagger : (P, p) \rightarrow (\varrho H, r)$ . By finality of  $\nu H$ , we have  $m \cdot e \cdot p^\dagger = x^\dagger \cdot q$  (with  $x^\dagger : (X, x) \rightarrow (\nu H, \tau)$ ). So the diagonal fill-in property induces a homomorphism  $(X, x) \rightarrow (B, b)$ , being the only homomorphism  $(X, x) \rightarrow (B, b)$  by the finality of  $\nu H$  and because  $m$  is monic.  $\square$

## 4 Instances of the Locally Finite Fixpoint

We will now present a number of instances of the LFF. First note, that all the known instances of the rational fixpoint (see e.g. [6, 12, 30]) are also instances of the locally finite fixpoint, because in all those cases the fp and fg objects coincide. For example, the class of regular languages is the rational fixpoint of  $2 \times (-)^\Sigma$  on  $\text{Set}$ . In this section, we will study further instances of the LFF that are most likely not instances of the rational fixpoint and which – to the best of our knowledge – have not been characterized by a universal property yet:

1. Behaviours of finite-state machines with side-effects as considered by the generalized powerset construction (cf. Sect. 4.1), particularly the following.
  - (a) Deterministic and ordinary context-free languages obtained as the behaviours of deterministic and non-deterministic stack-machines, respectively.
  - (b) Constructively  $S$ -algebraic formal power series, i.e. the “context-free” subclass of weighted languages with weights from a semiring  $S$ , yielded from weighted context-free grammars.
2. The monad of Courcelle’s algebraic trees.



### 4.1 Generalized Powerset Construction

The determinization of a non-deterministic automaton using the powerset construction is an instance of a more general framework, described by Silva, Bonchi, Bonsangue, and Rutten [41] based on an observation by Bartels [10] (see also Jacobs [26]). In that *generalized powerset construction*, an automaton with side-effects is turned into an ordinary automaton by internalizing the side-effects in the states. The LFF interacts well with this construction, because it precisely captures the behaviours of finite-state automata with side effects. The notion of side-effect is formalized by a monad, which induces the category, in which the LFF is considered.

In the following we assume that readers are familiar with monads and Eilenberg-Moore algebras (see e.g. [29] for an introduction). For a monad  $T$  on  $\mathcal{C}$  we denote by  $\mathcal{C}^T$  the category of Eilenberg-Moore algebras. Recall from [8, Corollary 2.75] that if  $\mathcal{C}$  is lfp (in most of our examples  $\mathcal{C}$  is **Set**) and  $T$  is finitary then  $\mathcal{C}^T$  is lfp, too, and for every fp object  $X$  the free Eilenberg-Moore algebra  $TX$  is fp in  $\mathcal{C}^T$ . In all the examples we consider below, the classes of fp and fg objects either provably differ or it is still unknown whether these classes coincide.

*Example 4.1.* In Sects. 4.4 and 4.5 we are going to make use of Moggi’s exception monad transformer (see e.g. [15]). Let us recall that for a fixed object  $E$ , the finitary functor  $(-) + E$  together with the unit  $\eta_X = \text{inl} : X \rightarrow X + E$  and multiplication  $\mu_X = \text{id}_X + [\text{id}_E, \text{id}_E] : X + E + E \rightarrow X + E$  form a finitary monad, the *exception monad*. Its algebras are  $E$ -pointed objects, i.e. objects  $X$ , together with a morphism  $E \rightarrow X$ , and homomorphisms are morphisms preserving the pointing. So the induced Eilenberg-Moore category is just the slice category  $\mathcal{C}^{(-)+E} \cong E/\mathcal{C}$ .

Now, given any monad  $T$  we obtain a new monad  $T(- + E)$  with obvious unit and multiplication. An Eilenberg-Moore algebra for  $T(- + E)$  consists of an Eilenberg-Moore algebra for  $T$  and an  $E$ -pointing, and homomorphisms are  $T$ -algebra homomorphisms preserving the pointing [25].

Now an automaton with side-effects is modelled as an  $HT$ -coalgebra, where  $T$  is a finitary monad on  $\mathcal{C}$  providing the type of side-effect. For example, for  $HX = 2 \times X^\Sigma$ , where  $\Sigma$  is an input alphabet,  $2 = \{0, 1\}$  and  $T$  the finite powerset monad on **Set**,  $HT$ -coalgebras are non-deterministic automata. However, the coalgebraic semantics using the final  $HT$ -coalgebra does not yield the usual language semantics of non-deterministic automata. To obtain this one considers the final coalgebra of a lifting of  $H$  to  $\mathcal{C}^T$ . Denote by  $U : \mathcal{C}^T \rightarrow \mathcal{C}$  the canonical forgetful functor.

**Definition 4.2.** *For a functor  $H : \mathcal{C} \rightarrow \mathcal{C}$  and a monad  $T : \mathcal{C} \rightarrow \mathcal{C}$ , a lifting of  $H$  is a functor  $H^T : \mathcal{C}^T \rightarrow \mathcal{C}^T$  such that  $H \cdot U = U \cdot H^T$ .*

If such a (not necessarily unique) lifting exists, the generalized powerset construction transforms an  $HT$ -coalgebra into a  $H^T$ -coalgebra on  $\mathcal{C}^T$ : For a coalgebra  $x : X \rightarrow HTX$ ,  $HTX$  carries an Eilenberg-Moore algebra, and one uses

freeness of the Eilenberg-Moore algebra  $TX$  to obtain a canonical  $T$ -algebra homomorphism  $x^\sharp : (TX, \mu^T) \rightarrow H^T(TX, \mu^T)$ . The *coalgebraic language semantics* of  $(X, x)$  is then given by  $X \xrightarrow{\eta_X} TX \xrightarrow{x^\sharp} \nu H^T$ , i.e. by composing the unique coalgebra morphism induced by  $x^\sharp$  with  $\eta_X$ . This construction yields a functor  $T' : \text{Coalg}(HT) \rightarrow \text{Coalg}H^T$  mapping coalgebras  $X \xrightarrow{x} HTX$  to  $x^\sharp$  and homomorphisms  $f$  to  $Tf$  (see e.g. [12, Proof of Lemma 3.27] for a proof).

Now our aim is to show that the LFF of  $H^T$  characterizes precisely the coalgebraic language semantics of all fp-carried  $HT$ -coalgebras. As the right adjoint  $U$  preserves monos and is faithful, we know that  $H^T$  preserves monos, and as  $T$  is finitary, filtered colimits in  $\mathcal{C}^T$  are created by the forgetful functor to  $\mathcal{C}$ , and we therefore see that  $H^T$  is finitary. Thus, by Theorem 3.8,  $\vartheta H^T$  exists and is a subcoalgebra of  $\nu H^T$ . By [37] and [10, Corollary 3.4.19], we know that  $\nu H^T$  is carried by  $\nu H$  equipped with a canonical algebra structure.

Now let us turn to the desired characterization of  $\vartheta H^T$ . Formally, the coalgebraic language semantics of all fp-carried  $HT$ -coalgebras is collected by forming the colimit  $k : K \rightarrow HK$  of the diagram  $\text{Coalg}_{\text{fp}} HT \xrightarrow{T'} \text{Coalg} H^T \xrightarrow{U} \text{Coalg} H$ . This coalgebra  $K$  is not yet a subcoalgebra of  $\nu H$  (for  $\mathcal{C} = \text{Set}$  that means, not all behaviourally equivalent states are identified in  $K$ ), but taking its image in  $\nu H$  we obtain the LFF:

**Proposition 4.3.** *The image  $(I, i)$  of the unique coalgebra morphism  $k^\dagger : K \rightarrow \nu H^T$  is precisely the locally finite fixpoint of the lifting  $H^T$ .*

One can also directly take the union of all desired behaviours, for  $\mathcal{C} = \text{Set}$ :

**Theorem 4.4.** *The locally finite fixpoint of the lifting  $H^T$  comprises precisely the images of determinized  $HT$ -coalgebras:*

$$\vartheta H^T = \bigcup_{\substack{x: X \rightarrow HTX \\ X \text{ finite}}} x^{\sharp\dagger}[TX] = \bigcup_{\substack{x: X \rightarrow HTX \\ X \text{ finite}}} x^{\sharp\dagger} \cdot \eta_X^T[X] \subseteq \nu H^T. \tag{1}$$

This result suggests that the locally finite fixpoint is the right object to consider in order to represent finite behaviour. We now instantiate the general theory with examples from the literature to characterize several well-known notions as LFF.

## 4.2 The Languages of Non-deterministic Automata

Let us start with a simple standard example. We already mentioned that non-deterministic automata are coalgebras for the functor  $X \mapsto 2 \times \mathcal{P}_f(X)^\Sigma$ . Hence they are  $HT$ -coalgebras for  $H = 2 \times (-)^\Sigma$  and  $T = \mathcal{P}_f$  the finite powerset monad on  $\text{Set}$ . The above generalized powerset construction then instantiates as the usual powerset construction that assigns to a given non-deterministic automaton its determinization.

Now note that the final coalgebra for  $H$  is carried by the set  $\mathcal{L} = \mathcal{P}(\Sigma^*)$  of all formal languages over  $\Sigma$  with the coalgebra structure given by  $o : \mathcal{L} \rightarrow 2$  with

$o(L) = 1$  iff  $L$  contains the empty word and  $t : \mathcal{L} \rightarrow \mathcal{L}^\Sigma$  with  $t(L)(s) = \{w \mid sw \in L\}$  the left language derivative. The functor  $H$  has a canonical lifting  $H^T$  on the Eilenberg-Moore category of  $\mathcal{P}_f$ , viz. the category of join semi-lattices. The final coalgebra  $\nu H^T$  is carried by all formal languages with the join semi-lattice structure given by union and  $\emptyset$  and with the above coalgebra structure. Furthermore, the coalgebraic language semantics of  $x : X \rightarrow HTX$  assigns to every state of the non-deterministic automaton  $X$  the language it accepts. Observe that join semi-lattices form a so-called *locally finite variety*, i.e. the finitely presentable algebras are precisely the finite ones. Hence, Theorem 4.4 states that the LFF of  $H^T$  is precisely the subcoalgebra of  $\nu H^T$  formed by all languages accepted by finite NFA, i.e. regular languages.

Note that in this example the LFF and the rational fixpoint coincide since both fp and fg join semi-lattices are simply the finite ones. Similar characterizations of the coalgebraic language semantics of finite coalgebras follow from Theorem 4.4 in other instances of the generalized powerset construction from [41] (cf. e.g. the treatment of the behaviour of finite weighted automata in [12]).

We now turn to examples that, to the best of our knowledge, cannot be treated using the rational fixpoint.

### 4.3 The Behaviour of Stack Machines

Push-down automata are finite state machines with infinitely many configurations. It is well-known that deterministic and non-deterministic pushdown automata recognize different classes of context-free languages. We will characterize both as instances of the locally finite fixpoint, using the results from [23] on stack machines, which can push or read *multiple* elements to or from the stack in a single transition, respectively.

That is, a transition of a stack machine in a certain state consists of reading an input character, going to a successor state based on the stack's topmost elements and of modifying the topmost elements of the stack. These stack operations are captured by the stack monad.

**Definition 4.5 (Stack monad, [22, Proposition 5]).** *For a finite set of stack symbols  $\Gamma$ , the stack monad is the submonad  $T$  of the store monad  $(- \times \Gamma^*)^{\Gamma^*}$  for which the elements  $\langle r, t \rangle$  of  $TX \subseteq (X \times \Gamma^*)^{\Gamma^*} \cong X^{\Gamma^*} \times (\Gamma^*)^{\Gamma^*}$  satisfy the following restriction: there exists  $k$  depending on  $r, t$  such that for every  $w \in \Gamma^k$  and  $u \in \Gamma^*$ ,  $r(wu) = r(w)$  and  $t(wu) = t(w)u$ .*

Note that the parameter  $k$  gives a bound on how many of the topmost stack cells the machine can access in one step.

Using the stack monad, stack machines are  $HT$ -coalgebras, where  $H = B \times (-)^\Sigma$  is the Moore automata functor for the finite input alphabet  $\Sigma$  and the set  $B$  of all predicates mapping (initial) stack configurations to output values from  $2$ , taking only the topmost  $k$  elements into account:  $B = \{p \in 2^{\Gamma^*} \mid \exists k \in \mathbb{N}_0 : \forall w, u \in \Gamma^*, |w| \geq k : p(wu) = p(w)\} \subseteq 2^{\Gamma^*}$ .

The final coalgebra  $\nu H$  is carried by  $B^{\Sigma^*}$  which is (modulo power laws) a set of predicates, mapping stack configurations to formal languages. Goncharov

et al. [23] show that  $H$  lifts to  $\mathbf{Set}^T$  and conclude that finite-state  $HT$ -coalgebras match the intuition of *deterministic* pushdown automata without spontaneous transitions. The languages accepted by those automata are precisely the *real-time deterministic context-free languages*; this notion goes back to Harrison and Havel [24]. We obtain the following, with  $\gamma_0$  playing the role of an initial symbol on the stack:

**Theorem 4.6.** *The locally finite fixpoint of  $H^T$  is carried by the set of all maps  $f \in B^{\Sigma^*}$  such that for any fixed  $\gamma_0 \in \Gamma$ ,  $\{w \in \Sigma^* \mid f(w)(\gamma_0) = 1\}$  is a real-time deterministic context-free language.*

*Proof.* By [23, Theorem 5.5], a language  $L$  is a real-time deterministic context-free language iff there exists some  $x : X \rightarrow HTX$ ,  $X$  finite, with its determination  $x^\sharp : TX \rightarrow HTX$  and there exist  $s \in X$  and  $\gamma_0 \in \Gamma$  such that  $f = x^\sharp \cdot \eta_X^T(s) \in B^{\Sigma^*}$  and  $f(w)(\gamma_0) = 1$  for all  $w \in \Sigma^*$ . The rest follows by (1).  $\square$

Just as for pushdown automata, the expressiveness of stack machines increases when equipping them with non-determinism. Technically, this is done by considering the *non-deterministic stack monad*  $T'$ , i.e.  $T'$  denotes a submonad of the non-deterministic store monad  $\mathcal{P}_f(- \times \Gamma^*)^{\Gamma^*}$ , as described in [23, Sect. 6]. In the non-deterministic setting, a similar property holds, namely that the determinized  $HT'$ -coalgebras with finite carrier describe precisely the context-free languages [23, Theorem 6.5]. Combine this with (1):

**Theorem 4.7.** *The locally finite fixpoint of  $H^{T'}$  is carried by the set of all maps  $f \in B^{\Sigma^*}$  such that for any fixed  $\gamma_0 \in \Gamma$ ,  $\{w \in \Sigma^* \mid f(w)(\gamma_0) = 1\}$  is a context-free language.*

#### 4.4 Context-Free Languages and Constructively $S$ -Algebraic Power Series

One generalizes from formal (resp. context-free) languages to weighted formal (resp. context-free) languages by assigning to each word a weight from a fixed semiring. More formally, a weighted language – a.k.a. *formal power series* – over an input alphabet  $X$  is defined as a map  $X^* \rightarrow S$ , where  $S$  is a semiring. The set of all formal power series is denoted by  $S\langle\langle X \rangle\rangle$ . Ordinary formal languages are formal power series over the boolean semiring  $\mathbb{B} = \{0, 1\}$ , i.e. maps  $X^* \rightarrow \{0, 1\}$ .

An important class of formal power series is that of *constructively  $S$ -algebraic* formal power series. We show that this class arises precisely as the LFF of the standard functor for deterministic Moore automata  $H = S \times (-)^\Sigma$ , but on an Eilenberg-Moore category of a  $\mathbf{Set}$  monad. As a special case, constructively  $\mathbb{B}$ -algebraic series are the context-free weighted languages and are precisely the LFF of the automata functor in the category of idempotent semirings.

The original definition of constructively  $S$ -algebraic formal power series goes back to Fliess [19], see also [17]. Here, we use the equivalent coalgebraic characterization by Winter et al. [44].

Let  $S\langle X \rangle \subseteq S\langle\langle X \rangle\rangle$  the subset of those maps, that are 0 for all but finitely many  $w \in X^*$ . If  $S$  is commutative, then  $S\langle - \rangle$  yields a finitary monad and

thus also  $T = S\langle - + \Sigma \rangle$  by Example 4.1. The algebras for  $S\langle - \rangle$  are associative  $S$ -algebras (over the commutative semiring  $S$ ), i.e.  $S$ -modules together with a monoid structure that is a module morphism in both arguments. The algebras for  $T$  are  $\Sigma$ -pointed  $S$ -algebras. The following notions are special instances of  $S$ -algebras.

*Example 4.8.* For  $S = \mathbb{B} = \{0, 1\}$ , one obtains idempotent semirings as  $\mathbb{B}$ -algebras, for  $S = \mathbb{N}$  semirings, and for  $S = \mathbb{Z}$  ordinary rings.

Winter et al. [44, Proposition 4] show that the final  $H$ -coalgebra is carried by  $S\langle\langle \Sigma \rangle\rangle$  and that constructively  $S$ -algebraic series are precisely those elements of  $S\langle\langle \Sigma \rangle\rangle$  that arise as the behaviours of those coalgebra  $c : X \rightarrow HS\langle X \rangle$  with finite  $X$ , after determinizing them to some  $c^\sharp : S\langle X \rangle \rightarrow HS\langle X \rangle$  (see [44, Theorem 23]).

However, this determinization is not directly an instance of the generalized powerset construction. We shall show that the same behaviours can be obtained by using the standard generalized powerset construction with an appropriate lifting of  $H$  to  $T$ -algebras. Having an  $S$ -algebra structure on  $A$  and a  $\Sigma$ -pointing  $j : \Sigma \rightarrow A$  we need to define another  $S$ -algebra structure and  $\Sigma$ -pointing on  $HA = S \times A^\Sigma$ . While the  $S$ -module structure is just point-wise, we need to take care when multiplying two elements from  $HA$ . To this end we first we define the operation  $[-, -] : S \times A^\Sigma \rightarrow A$  by

$$[o, \delta] := i(o) + \sum_{b \in \Sigma} (j(b) \cdot \delta(b)),$$

where  $i : S \rightarrow A$  is the canonical map with  $i(s) = s \cdot 1$  with 1 the neutral element of the monoid on  $A$ . The idea is that  $[o, \delta]$  acts like a state with output  $o$  and derivation  $\delta$ . The multiplication on  $HA = S \times A^\Sigma$  is then defined by

$$(o_1, \delta_1) * (o_2, \delta_2) := (o_1 \cdot o_2, a \mapsto \delta_1(a) \cdot [o_2, \delta_2] + i(o_1) \cdot \delta_2(a)). \tag{2}$$

The  $\Sigma$ -pointing is the obvious:  $a \mapsto (0, \varrho_a)$  where  $\varrho_a(a) = 1$  and  $\varrho_a(b) = 0$  for  $a \neq b$ .

**Lemma 4.9.** *For any  $w \in A$  in  $\mathbf{Set}^T$  and any  $H^T$ -coalgebra structure  $c : A \rightarrow H^T A$ ,  $w$  and  $[c(w)]$  are behaviourally equivalent in  $\mathbf{Set}$ .*

Given a coalgebra  $c : X \rightarrow HS\langle X \rangle$ , Winter et al. [44, Proposition 14] determinize  $c$  to some  $\hat{c} = \langle \hat{o}, \hat{\delta} \rangle : S\langle X \rangle \rightarrow HS\langle X \rangle$  with the property that for any  $v, w \in S\langle X \rangle$ ,

$$\hat{o}(v * w) = \hat{o}(v) \cdot \hat{o}(w) \quad \text{and} \quad \hat{\delta}(v * w, a) = \hat{\delta}(v, a) * w + \hat{o}(v) * \hat{\delta}(w, a), \tag{3}$$

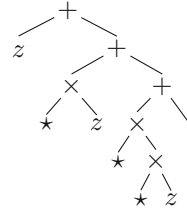
and such that  $\hat{c}$  is a  $S$ -module homomorphism. However, the generalized powerset construction w.r.t.  $T$  yields a coalgebra  $c^\sharp : S\langle X + \Sigma \rangle \rightarrow HS\langle X + \Sigma \rangle$ . The above property, together with Lemma 4.9 and (2) implies that  $\hat{c}$  and  $c^\sharp$  are essentially the same coalgebra structures:

**Lemma 4.10.** *In  $\mathbf{Set}$ ,  $u \in (S\langle X \rangle, \hat{c})$  and  $S\langle \text{inl} \rangle(u) \in (S\langle X + \Sigma \rangle, c^\sharp)$  are behaviourally equivalent.*

It follows that  $\hat{c}^\dagger = c^\sharp \cdot S\langle \text{inl} \rangle$  and thus their images in  $\nu H$  are identical. Hence, a formal power series is constructively  $S$ -algebraic iff it is in the image of some  $c^\sharp \cdot S\langle \text{inl} \rangle$ , and by (1), iff it is in the locally finite fixpoint of  $H^T$ .

### 4.5 Courcelle’s Algebraic Trees

For a fixed signature  $\Sigma$  of so called *givens*, a *recursive program scheme* (or *rps*, for short) contains mutually recursive definitions of new operations  $\varphi_1, \dots, \varphi_k$  (with respective arities  $n_1, \dots, n_k$ ). The recursive definition of  $\varphi_i$  may involve symbols from  $\Sigma$ , operations  $\varphi_1, \dots, \varphi_k$  and  $n_i$  variables  $x_1, \dots, x_{n_i}$ . The (uninterpreted) solution of an rps is obtained by unravelling these recursive definitions, producing a possibly infinite  $\Sigma$ -tree over  $x_1, \dots, x_{n_i}$  for each operation  $\varphi_i$ . Figure 1 shows an rps over the signature  $\Sigma = \{\star/0, \times/2, +/2\}$  and its solution. In general, an *algebraic  $\Sigma$ -tree* is a  $\Sigma$ -tree which is definable by an rps over  $\Sigma$  (see Courcelle [16]).



**Fig. 1.** Solution of  $\varphi(z) = z + \varphi(\star \times z)$

Generalizing from a signature to a finitary endofunctor  $H : \mathcal{C} \rightarrow \mathcal{C}$  on an lfp category, Adámek et al. [7] describe an rps as a coalgebra for a functor  $\mathcal{H}_f$  on  $H/\mathbf{Mnd}_f(\mathcal{C})$ , in which objects are finitary  $H$ -pointed monads on  $\mathcal{C}$ , i.e. finitary monads  $M$  together with a natural transformation  $H \rightarrow M$ . They introduce the *context-free monad*  $C^H$  of  $H$ , which is an  $H$ -pointed monad that is a subcoalgebra of the final coalgebra for  $\mathcal{H}_f$  and which is the monad of Courcelle’s algebraic  $\Sigma$ -trees in the special case where  $\mathcal{C} = \mathbf{Set}$  and  $H$  is a polynomial functor associated to a signature  $\Sigma$ . We will prove that this monad is the LFF of  $\mathcal{H}_f$ , and thereby we characterize it by a universal property – solving the open problem in [7].

The setting is again an instance of the generalized powerset construction, but this time with  $\mathbf{Fun}_f(\mathcal{C})$  as the base category in lieu of  $\mathbf{Set}$ . Let  $\mathcal{C}$  be an lfp category in which the coproduct injections are monic and consider a finitary, mono-preserving endofunctor  $H : \mathcal{C} \rightarrow \mathcal{C}$ . Denote by  $\mathbf{Fun}_f(\mathcal{C})$  the category of finitary endofunctors on  $\mathcal{C}$ . Then  $H$  induces an endofunctor  $H \cdot (-) + \text{Id}$  on  $\mathbf{Fun}_f(\mathcal{C})$ , denoted  $\dot{H}$  and mapping an endofunctor  $V$  to the functor  $X \mapsto HVX + X$ . This functor  $\dot{H}$  gets precomposed with a monad on  $\mathbf{Fun}_f(\mathcal{C})$  as we now explain.

**Proposition 4.11 (Free monad, [5, 9]).** *For a finitary endofunctor  $H$ , free  $H$ -algebras  $\varphi_X : HF^H X \rightarrow F^H X$  exist for all  $X \in \mathcal{C}$ .  $F^H$  itself is a finitary monad on  $\mathcal{C}$ , more specifically it is the free monad on  $H$ .*

For example, if  $H$  is a polynomial functor associated to a signature  $\Sigma$ , then  $F^H X$  is the usual term algebra that contains all finite  $\Sigma$ -trees over the set of generators  $X$ . Proposition 4.11 implies that  $H \mapsto F^H$  is the object assignment of a monad on  $\mathbf{Fun}_f(\mathcal{C})$ . The Eilenberg-Moore category of  $F^{(-)}$  is easily seen to be  $\mathbf{Mnd}_f(\mathcal{C})$ , the category of finitary monads on  $\mathcal{C}$ . Here, fp and fg objects differ, see [45, Sect. 5.4.1] for a proof.

Similarly as in the case of context-free languages, we will work with the monad  $E^{(-)} = F^{H+(-)}$ , so we get  $H$ -pointed finitary monads as the  $E^{(-)}$ -algebras. This category is equivalent to a slice category: the universal property induced by  $F^{(-)}$  states, that for any finitary monad  $B$  the natural transformations  $H \rightarrow B$  are in one-to-one correspondence with monad morphisms  $F^H \rightarrow B$ ; so the category  $H/\text{Mnd}_f(\mathcal{C})$  of finitary  $H$ -pointed monads on  $\mathcal{C}$  is isomorphic to the slice category  $F^H/\text{Mnd}_f(\mathcal{C})$ . This finishes the description of the base category and we now lift the functor  $\dot{H}$  to this category.

Consider an  $H$ -pointed monad  $(B, \beta : H \rightarrow \mathcal{C}) \in H/\text{Mnd}_f(\mathcal{C})$ . By [21], the endofunctor  $H \cdot B + \text{Id}$  carries a canonical monad structure. Furthermore, we have an obvious pointing  $\text{inl} \cdot H\eta^B : H \rightarrow H \cdot B + \text{Id}$ . By [32], this defines an endofunctor on  $H$ -pointed monads,  $\mathcal{H}_f : H/\text{Mnd}_f(\mathcal{C}) \rightarrow H/\text{Mnd}_f(\mathcal{C})$ , which is a lifting of  $\dot{H}$ . In order to verify that  $\mathcal{H}_f$  is finitary, we first need to know how filtered colimits look in  $H/\text{Mnd}_f(\mathcal{C})$ .

**Lemma 4.12.** *The forgetful  $U : \text{Mnd}_f(\mathcal{C}) \rightarrow \text{Fun}_f(\mathcal{C})$  creates filtered colimits.*

Clearly, the canonical projection functor  $H/\text{Mnd}_f(\mathcal{C}) \rightarrow \text{Mnd}_f(\mathcal{C})$  creates filtered colimits, too. Therefore, filtered colimits in the slice category  $H/\text{Mnd}_f(\mathcal{C})$  are formed on the level of  $\text{Fun}_f(\mathcal{C})$ , i.e. object-wise. The functor  $\dot{H}$  is finitary on  $\text{Fun}_f(\mathcal{C})$  and thus also its lifting  $\mathcal{H}_f$  is finitary. So all requirements from Assumption 3.1 are met: we have a finitary endofunctor  $\mathcal{H}_f$  on the  $\text{lf}$  category  $H/\text{Mnd}_f(\mathcal{C})$ , and by [7, Corollary 2.20]  $\mathcal{H}_f$  preserves monos since  $H$  does. By Theorem 3.8,  $\mathcal{H}_f$  has a locally finite fixpoint.

*Remark 4.13.* The final  $\mathcal{H}_f$ -coalgebra is not of much interest, but that of a related functor.  $\mathcal{H}_f$  generalizes to a functor  $\mathcal{H} : H/\text{Mnd}_c(\mathcal{C}) \rightarrow H/\text{Mnd}_c(\mathcal{C})$  on  $H$ -pointed countably accessible<sup>1</sup> monads. For any object  $X \in \mathcal{C}$ , the finitary endofunctor  $H(-) + X$  has a final coalgebra; call the carrier  $TX$ . Then  $T$  is a monad [1], is countably accessible [7] and is the final  $\mathcal{H}$ -coalgebra [32].

Adámek et al. [7] characterize a (guarded) recursive program scheme as a natural transformation  $V \rightarrow H \cdot E^V + \text{Id}$  with  $V$  fp (in  $\text{Fun}_f(\mathcal{C})$ ), or equivalently, via the generalized powerset construction w.r.t. the monad  $E^{(-)}$  as an  $\mathcal{H}_f$ -coalgebra on the carrier  $E^V$  (in  $\text{Mnd}_f(\mathcal{C})$ ). These  $\mathcal{H}_f$ -coalgebras on carriers  $E^V$  where  $V \in \text{Fun}_f(\mathcal{C})$  is fp form the full subcategory  $\text{EQ} \subseteq \text{Coalg}\mathcal{H}_f$ . They show two equivalent ways of constructing the monad of Courcelle’s algebraic trees for the case  $\mathcal{C} = \text{Set}$ : as the image of  $\text{EQ}_{\text{colim}}$  in the final coalgebra  $T$  of Remark 4.13, and as the colimit of  $\text{EQ}_2$ , where  $\text{EQ}_2$  is the closure of  $\text{EQ}$  under strong quotients. We now provide a third characterization, and show that the monad of Courcelle’s algebraic trees is the locally finite fixpoint of  $\mathcal{H}_f$ .

To this end it suffices to show that  $\text{EQ}_2$  is precisely the diagram of  $\mathcal{H}_f$ -coalgebras with an fg carrier. This is established with the help of the following two technical lemmas. We now assume that  $\mathcal{C} = \text{Set}$ .

---

<sup>1</sup> A colimit is *countably filtered* if its diagram has for every countable subcategory a cocone. A functor is *countably accessible* if it preserves countably filtered colimits.

**Lemma 4.14.**  $\mathcal{H}_f$  maps strong epis to morphisms carried by strong epi natural transformations.

We have the following variation of Proposition 3.17:

**Lemma 4.15.** Any  $\mathcal{H}_f$ -coalgebra  $b : (B, \beta) \rightarrow \mathcal{H}_f(B, \beta)$ , with  $B$  fg, is the strong quotient of a coalgebra from EQ.

The proof of Lemma 4.15 makes use of Lemma 4.14 as well as the following properties:

- The fp objects in  $\text{Fun}_f(\text{Set})$  are the quotients of polynomial functors.
- The polynomial functors are projective. That means that for a polynomial functor  $P$  and any natural transformation  $n : K \rightarrow L$  with surjective components we have the following property: for every  $f : P \rightarrow L$  there exists  $f' : P \rightarrow K$  with  $n \cdot f' = f$ .
- Any fg object in  $H/\text{Mnd}_f(\text{Set})$  is the quotient of some  $E^V$  with  $V$  fp in  $\text{Fun}_f(\text{Set})$  and thus also of some  $E^P$  with  $P$  a polynomial functor.

Note that the last property holds because  $H/\text{Mnd}_f(\text{Set})$  is an Eilenberg-Moore category and  $E^V$  is the free Eilenberg-Moore algebra on the fp object  $V$ . It follows from Lemma 4.15 that  $\text{Coalg}_{\text{fg}} \mathcal{H}_f$  is the same category as  $\text{EQ}_2$ ; thus their colimits in  $\text{Coalg} \mathcal{H}_f$  are isomorphic and we conclude:

**Theorem 4.16.** The locally finite fixpoint of  $\mathcal{H}_f : H_\Sigma/\text{Mnd}_f(\text{Set}) \rightarrow H_\Sigma/\text{Mnd}_f(\text{Set})$  is the monad of Courcelle's algebraic trees, sending a set to the algebraic  $\Sigma$ -trees over it.

## 5 Conclusions and Future Work

We have introduced the locally finite fixpoint of a finitary mono-preserving endofunctor on an lfp category. We proved that this fixpoint is characterized by two universal properties: it is the final lfg coalgebra and the initial fg-iterative algebra for the given endofunctor. And we have seen many instances where the LFF is the domain of behaviour of finite-state and finite-equation systems. In particular all previously known instances of the rational fixpoint are also instances of the LFF, and we have obtained a number of interesting further instances not captured by the rational fixpoint.

On a more technical level, the LFF solves a problem that sometimes makes the rational fixpoint hard to apply. The latter identifies behaviourally equivalent states (i.e. is a subcoalgebra of the final coalgebra) if the classes of fp and fg objects coincide. This condition, however, may be false or unknown (and sometimes non-trivial to establish) in a given lfp category. But the LFF always identifies behaviourally equivalent states.

There are a number of interesting topics for future work concerning the LFF. First, it should be interesting to obtain further instances of the LFF, e.g. analyzing the behaviour of tape machines [23] may perhaps lead to a description of the



recursively enumerable languages by the LFF. Second, syntactic descriptions of the LFF are of interest. In works such as [12, 35, 40, 42] Kleene type theorems and axiomatizations of the behaviour of finite systems are studied. Completeness of an axiomatization is then established by proving that expressions modulo axioms form the rational fixpoint. It is an interesting question whether the theory of the LFF we presented here may be of help as a tool for syntactic descriptions and axiomatizations of further system types.

As we have mentioned already the rational fixpoint is the starting point for the coalgebraic study of iterative and iteration theories. A similar path could be followed based on the LFF and this should lead to new coalgebraic iteration/recursion principles, in particular in instances such as context-free languages or constructively  $S$ -algebraic formal power series.

Another approach to more powerful recursive definition principles are abstract operational rules (see [27] for an overview). It has been shown that certain rule formats define operations on the rational fixpoint [13, 31], and it should be investigated whether a similar theory can be developed based on the LFF.

Finally, in the special setting of Eilenberg-Moore categories one could base the study of finite systems on *free* finitely generated algebras (rather than all fp or all fg algebras). Does this give a third fixpoint capturing behaviour of finite state systems with side effects besides the rational fixpoint and the LFF? And what is then the relation between the three fixpoints? Also the parallelism in the technical development between rational fixpoint and LFF indicates that there should be a general theory that is parametric in a class of “finite objects” and that allows to obtain results about the rational fixpoint, the LFF and other possible “finite behaviour domains” as instances.

## References

1. Aczel, P., Adámek, J., Milius, S., Velebil, J.: Infinite trees and completely iterative theories: a coalgebraic view. *Theoret. Comput. Sci.* **300**, 1–45 (2003)
2. Adámek, J., Milius, S., Velebil, J.: Semantics of higher-order recursion schemes. *Log. Methods Comput. Sci.* **7**(1:15), 43 (2011)
3. Adámek, J., Milius, S., Velebil, J.: Equational properties of iterative monads. *Inf. Comput.* **208**, 1306–1348 (2010)
4. Adámek, J., Milius, S., Velebil, J.: Elgot theories: a new perspective of the equational properties of iteration. *Math. Struct. Comput. Sci.* **21**(2), 417–480 (2011)
5. Adámek, J.: Free algebras and automata realizations in the language of categories. *Comment. Math. Univ. Carolin.* **015**(4), 589–602 (1974)
6. Adámek, J., Milius, S., Velebil, J.: Iterative algebras at work. *Math. Struct. Comput. Sci.* **16**(6), 1085–1131 (2006)
7. Adámek, J., Milius, S., Velebil, J.: On second-order iterative monads. *Theoret. Comput. Sci.* **412**(38), 4969–4988 (2011)
8. Adámek, J., Rosický, J.: *Locally Presentable and Accessible Categories*. Cambridge University Press, New York (1994)
9. Barr, M.: Coequalizers and free triples. *Math. Z.* **116**, 307–322 (1970)

10. Bartels, F.: On generalized coinduction and probabilistic specification formats: Distributive laws in coalgebraic modelling. Ph.D. thesis, Vrije Universiteit Amsterdam.(2004)
11. Bloom, S.L., Ésik, Z.: Iteration Theories: The Equational Logic of Iterative Processes. EATCS Monographs on Theoretical Computer Science. Springer, Heidelberg (1993)
12. Bonsangue, M., Milius, S., Silva, A.: Sound and complete axiomatizations of coalgebraic language equivalence. *ACM Trans. Comput. Log.* 14((1: 7)), 52.(2013)
13. Bonsangue, M.M., Milius, S., Rot, J.: On the specification of operations on the rational behaviour of systems. In: Luttkik, B., Reniers, M.A. (eds.) Proceedings of Combined Workshop on Expressiveness in Concurrency and Structural Operational Semantics (EXPRESS/SOS'12), Electronic Proceedings of Theoretical Computer Science, vol. 89, pp. 3–18.(2012)
14. Borceux, F.: Handbook of Categorical Algebra: Volume 1, Basic Category Theory. Encyclopedia of Mathematics and its Applications. Cambridge University Press, Cambridge (1994)
15. Cenciarelli, P., Moggi, E.: A syntactic approach to modularity in denotational semantic. In: Proceedings of 5th CTCS. CWI Technical report.(1993)
16. Courcelle, B.: Fundamental properties of infinite trees. *Theoret. Comput. Sci.* **25**, 95–169 (1983)
17. Droste, M., Kuich, W., Vogler, H.: Handbook of Weighted Automata, 1st edn. Springer Publishing Company, Berlin (2009)
18. Elgot, C.: Monadic computation and iterative algebraic theories. In: Rose, H.E., Sheperdson, J.C. (eds.) Logic Colloquium 1973, vol. 80, pp. 175–230. North-Holland Publishers, Amsterdam (1975)
19. Fliess, M.: Sur divers produits de séries formelles. *Bulletin de la Société Mathématique de France* **102**, 181–191 (1974)
20. Gabriel, P., Ulmer, F.: Lokal präsentierbare Kategorien. Lecture Notes in Mathematics, vol. 221. Springer, Heidelberg (1971)
21. Ghani, N., Lüth, C., Marchi, F.D.: Monads of coalgebras: rational terms and term graphs. *Math. Struct. Comput. Sci.* **15**, 433–451 (2005)
22. Goncharov, Sergey: Trace Semantics via Generic Observations. In: Heckel, Reiko, Milius, Stefan (eds.) CALCO 2013. LNCS, vol. 8089, pp. 158–174. Springer, Heidelberg (2013)
23. Goncharov, Sergey, Milius, Stefan, Silva, Alexandra: Towards a Coalgebraic Chomsky Hierarchy. In: Diaz, Josep, Lanese, Ivan, Sangiorgi, Davide (eds.) TCS 2014. LNCS, vol. 8705, pp. 265–280. Springer, Heidelberg (2014)
24. Harrison, M.A., Havel, I.M.: Real-time strict deterministic languages. *SIAM J. Comput.* **1**(4), 333–349 (1972)
25. Hyland, M., Plotkin, G., Power, J.: Combining effects: sum and tensor. *Theoret. Comput. Sci.* **357**(1–3), 70–99 (2006)
26. Jacobs, Bart: A Bialgebraic Review of Deterministic Automata, Regular Expressions and Languages. In: Futatsugi, Kokichi, Jouannaud, Jean-Pierre, Meseguer, José (eds.) Algebra, Meaning, and Computation. LNCS, vol. 4060, pp. 375–404. Springer, Heidelberg (2006)
27. Klin, B.: Bialgebras for structural operational semantics: an introduction. *Theoret. Comput. Sci.* **412**(38), 5043–5069 (2011)
28. Lambek, J.: A fixpoint theorem for complete categories. *Math. Z.* **103**, 151–161 (1968)
29. MacLane, S.: Categories for the Working Mathematician. Graduate Texts in Mathematics, 2nd edn. Springer, New York.(1998)

30. Milius, S.: A sound and complete calculus for finite stream circuits. In: Proceedings of 25th Annual Symposium on Logic in Computer Science (LICS 2010), pp. 449–458.(2010)
31. Milius, S., Bonsangue, M.M., Myers, R.S., Rot, J.: Rational operation models. *Electron. Notes Theoret. Comput. Sci.* 298, 257–282. In: Mislove, M. (ed.) Proceedings of 29th conference on Mathematical Foundations of Programming Science (MFPS XXIX).(2013)
32. Milius, S., Moss, L.S.: The category theoretic solution of recursive program schemes. *Theoret. Comput. Sci.* **366**, 3–59 (2006)
33. Milius, S., Pattinson, D., Wißmann, T.: A new foundation for finitary corecursion: the locally finite fixpoint and its properties (2015).. <http://arxiv.org/abs/1601.01532>
34. Milius, S., Wißmann, T.: Finitary corecursion for the infinitary lambda calculus. In: Proceedings of 6th Conference on Algebra and Coalgebra in Computer Science, CALCO 2015. Leibniz International Proceedings in Informatics.(2015)
35. Myers, R.: Rational coalgebraic machines in varieties: Languages, completeness and automatic proofs. Ph.D. thesis, Imperial College London, Department of Computing.(2011)
36. Petre, I., Salomaa, A.: Algebraic systems and pushdown automata. In: Droste, M., Kuich, W., Vogler, H. (eds.) *Handbook of Weighted Automata*. Monographs in Theoretical Computer Science. An EATCS Series, pp. 257–289.. Springer, Heidelberg (2009)
37. Plotkin, G., Turi, D.: Towards a mathematical operational semantics. In: Proceedings of 12th LICS Conference, pp. 280–291. IEEE Computer Society Press.(1997)
38. Rutten, J.J.M.M.: Automata and Coinduction. In: Sangiorgi, Davide, de Simone, Robert (eds.) *CONCUR 1998*. LNCS, vol. 1466, pp. 194–218. Springer, Heidelberg (1998)
39. Rutten, J.: Universal coalgebra: a theory of systems. *Theoret. Comput. Sci.* **249**(1), 3–80 (2000)
40. Silva, A., Bonchi, F., Bonsangue, M.M., Rutten, J.J.M.M.: Quantitative kleene coalgebras. *Inf. Comput.* **209**(5), 822–849 (2011)
41. Silva, A., Bonchi, F., Bonsangue, M.M., Rutten, J.: Generalizing determinization from automata to coalgebras. *Log. Meth. Comput. Sci.* **9**(1), 1–27 (2013)
42. Silva, A., Bonsangue, M.M., Rutten, J.: Non-deterministic Kleene coalgebras. *Log. Meth. Comput. Sci.* 6(3: 23), 39.(2010)
43. Winter, J., Bonsangue, M., Rutten, J.: Coalgebraic characterizations of context-free languages. *Log. Meth. Comput. Sci.* **9**(3), 1–39 (2013)
44. Winter, J., Bonsangue, M.M., Rutten, J.J.: Context-free coalgebras. *J. Comput. Syst. Sci.* **81**(5), 911–939 (2015)
45. Wißmann, T.: The locally finite fixpoint and its properties. Master’s thesis, Friedrich-Alexander Universität Erlangen-Nürnberg (April 2015). <http://thorsten-wissmann.de/theses/ma-wissmann.pdf>