Completeness for Categories of Generalized Automata

((Co)algebraic pearls) / CALCO 2023

Guido Boccali[†], Andrea Laretto[⊛], Fosco Loregian[⊛], Stefano Luneia[‡]

2023-06-21

[†] : Università di Torino
[⊛] : Tallinn University of Technology
[‡] : Università di Bologna

Context

- Many ways of *categorifying* automata theory in category theory [Adámek-Trnková, 1990], [Rutten, 2000], [Jacobs, 2006]
- Very long tradition, with some works already from the 1970s [Ehrig et al. 1974], [Naudé, 1977], [Guitart, 1980]
- We want to study the *completeness of categories of F-automata*
- This is known in the literature for the case $F := - \otimes I$ [Ehrig, 1974]: *we present a generalization based on a more conceptual approach.*
- We formalize some of our results in the Agda proof assistant.

Automata in monoidal categories

- Our setting: *automata in a monoidal category* $(\mathcal{K}, \otimes, 1)$
- Take two fixed $I, O \in \mathcal{K}$, representing input and output objects.

Definition

A **Moore automata** $\langle E, d, s \rangle$ in $\mathcal{K}$ is an object E with morphisms d, s:

$$E \xleftarrow{d} E \otimes I \;\; ; \;\; E \xrightarrow{s} O$$

Definition

A **morphism of Moore automata** between $\langle E, d, s \rangle$ and $\langle T, d', s' \rangle$ is a morphism $f \colon E \to T$ making the following diagrams commute:

$$\begin{array}{ccc} E & \xleftarrow{d} & E \otimes I \\ {\scriptstyle f}\downarrow & & \downarrow{\scriptstyle f \otimes I} \\ T & \xleftarrow[d']{} & T \otimes I \end{array} \qquad \begin{array}{ccc} E & \xrightarrow{s} & O \\ {\scriptstyle f}\downarrow & & \| \\ T & \xrightarrow[s']{} & O \end{array}$$

- We denote the category of Moore automata as $\mathrm{Moore}(I, O)$.

Mealy automata

Definition

A **Mealy automata** in $\mathcal{K}$ is a span of two morphisms d and s:

$$E \xleftarrow{\ d\ } E \otimes I \xrightarrow{\ s\ } O$$

Definition

A **morphism of Mealy automata** between $\langle E, d, s \rangle$ and $\langle T, d', s' \rangle$ is a morphism $f\colon E \to T$ making the following diagram commute:

$$\begin{array}{ccccc} E & \xleftarrow{\ d\ } & E \otimes I & \xrightarrow{\ s\ } & O \\ {\scriptstyle f}\downarrow & & \downarrow{\scriptstyle f \otimes I} & & \Vert \\ T & \xleftarrow[\ d'\ ]{} & T \otimes I & \xrightarrow[\ s'\ ]{} & O \end{array}$$

- Mealy automata arrange into categories $\mathrm{Mealy}(I, O)$, which are actually the hom-categories of a bicategory![1]
  [1]: *[Boccali, Laretto, Loregian, Luneia, 2023]*

- A natural generalization: replace $- \otimes I$ in the domain with a generic endofunctor $F : \mathcal{K} \to \mathcal{K}$ acting on the states.
- Idea: imagine F as providing an action context for the automaton.

Definition

A **morphism of F-automata** between $\langle E, d, s \rangle$ and $\langle T, d', s' \rangle$ is a morphism $f : E \to T$ making the following diagrams commute:

$$\text{Moore: } \begin{array}{ccc} E & \xleftarrow{d} & FE \\ {\scriptstyle f}\downarrow & & \downarrow{\scriptstyle Ff} \\ T & \xleftarrow[d']{} & FT \end{array} \quad \begin{array}{ccc} E & \xrightarrow{s} & O \\ {\scriptstyle f}\downarrow & & \| \\ T & \xrightarrow[s']{} & O \end{array} \qquad \text{Mealy: } \begin{array}{ccccc} E & \xleftarrow{d} & FE & \xrightarrow{s} & O \\ {\scriptstyle f}\downarrow & & \downarrow{\scriptstyle Ff} & & \| \\ T & \xleftarrow[d']{} & FT & \xrightarrow[s']{} & O \end{array}$$

- We denote the category of F-Moore automata by $F\text{–}\mathrm{Moore}(O)$.
- A Moore machine is an F-Moore machine where $F : \mathcal{K} \to \mathcal{K}$ is the functor $- \otimes I := E \mapsto E \otimes I$.

Examples of F

- Different choices of F lead to different notions of automata (e.g., sequential/tree/linear automata) [Adámek-Trnková, 1990].
- In particular, we take into consideration the case where F **has a right adjoint** R.

$$E \xleftarrow{d} FE \xrightarrow{s} O,$$

$$RE \xleftarrow{d} E \xrightarrow{s} RO.$$

- In the case where $F := I \otimes -$, this corresponds to the internal hom $R := [I, -]$, thus associating a ”transition map” to each state.

Completeness of categories of categorical automata

- We are interested in the following question:

  When is the category of F-automata (co)complete?

- Our contribution: a conceptual proof that $F\text{–}\mathrm{Moore}(O)$ and $F\text{–}\mathrm{Mealy}(O)$ are (co)complete when $\mathcal{K}$ is, based on the theory of *2-pullbacks* in Cat and basic facts about limit-preserving functors.
- Proof sketch:
  1. Present $F\text{–}\mathrm{Moore}(O)$ and $F\text{–}\mathrm{Mealy}(O)$ as 2-pullbacks in Cat.
  2. Theorem [Mac Lane, 1998]: if the functors of the pullback satisfy some conditions, then we can compute limits in the pullback.
  3. The functors characterizing $F\text{–}\mathrm{Moore}(O)$ and $F\text{–}\mathrm{Mealy}(O)$ satisfy the conditions, along with the fact that F is a left adjoint.
  4. Hence, the categories are complete when the base category $\mathcal{K}$ also is.

Characterization of F-Moore and F-Mealy

Theorem

The categories $F\text{–}\mathrm{Moore}(O)$ and $F\text{–}\mathrm{Mealy}(O)$ can be characterized as the following strict 2-pullbacks in Cat:

$$\begin{array}{ccc} F\text{–}\mathrm{Moore}(O) & \xrightarrow{U} & (\mathcal{K}_{/O}) \\ {\scriptstyle V}\downarrow & \lrcorner & \downarrow{\scriptstyle dom} \\ \mathrm{Alg}(F) & \xrightarrow[forget]{} & \mathcal{K} \end{array} \qquad \begin{array}{ccc} F\text{–}\mathrm{Mealy}(O) & \xrightarrow{U} & (F \downarrow O) \\ {\scriptstyle V}\downarrow & \lrcorner & \downarrow{\scriptstyle dom} \\ \mathrm{Alg}(F) & \xrightarrow[forget]{} & \mathcal{K} \end{array}$$

- $\mathrm{Alg}(F)$ *is the category of algebras of F in $\mathcal{K}$.*
- ***forget*** *is the canonical forgetful functor of F-algebras.*
- $(\mathcal{K}_{/O})$ *is the slice category of $\mathcal{K}$ over O.*
- ***dom*** *is the forgetful functor of comma categories on the domain.*
- $(F \downarrow O)$ *is the comma category defined by F and the constant functor on the object O.*

Intuition, F-Mealy as pullback

- Intution for the characterization of F–$\mathrm{Mealy}(O)$ as pullback in Cat:

$$\begin{array}{ccccc} E & \xleftarrow{\;d\;} & FE & \xrightarrow{\;s\;} & O \\ {\scriptstyle f}\big\downarrow & & \big\downarrow{\scriptstyle Ff} & & \big\| \\ T & \xleftarrow[\;d'\;]{} & FT & \xrightarrow[\;s'\;]{} & O \end{array}$$

$$\swarrow \qquad\qquad\qquad\qquad \searrow$$

$$\mathrm{Alg}(F) \quad \begin{array}{ccc} E & \xleftarrow{\;d\;} & FE \\ {\scriptstyle f}\big\downarrow & & \big\downarrow{\scriptstyle Ff} \\ T & \xleftarrow[\;d'\;]{} & FT \end{array} \quad \rightarrow \quad \begin{array}{c} E \\ {\scriptstyle f}\big\downarrow \\ T \end{array} \quad \leftarrow \quad \begin{array}{ccc} FE & \xrightarrow{\;s\;} & O \\ {\scriptstyle Ff}\big\downarrow & & \big\| \\ FT & \xrightarrow[\;s'\;]{} & O \end{array} \quad (F \downarrow O)$$

Basic notions on limits

Definition

A functor $F : \mathcal{A} \to \mathcal{B}$ **preserves** limits of shape $J : \mathcal{I} \to \mathcal{A}$ when, given a limit x in $\mathcal{A}$, then $F(x)$ is the limit of the composite diagram $\mathcal{I} \xrightarrow{J} \mathcal{A} \xrightarrow{F} \mathcal{B}$.

Definition

A functor $F : \mathcal{A} \to \mathcal{B}$ **reflects** limits of shape $J : \mathcal{I} \to \mathcal{A}$ when, given a cone x in $\mathcal{A}$ such that $F(x)$ is the limit of the composite diagram $\mathcal{I} \xrightarrow{J} \mathcal{A} \xrightarrow{F} \mathcal{B}$, then x was already a limit of J in $\mathcal{A}$.

Definition

A functor $F : \mathcal{A} \to \mathcal{B}$ **creates** limits of shape $J : \mathcal{I} \to \mathcal{A}$ when it both **preserves** and **reflects** them.

Pullbacks in Cat and limits

Theorem (Mac Lane 1998, V.6, Ex. 3)

Given a pullback diagram in Cat*:*

$$\begin{array}{ccc} \mathcal{A} & \xrightarrow{H'} & \mathcal{Y} \\ {\scriptstyle G'}\downarrow & \lrcorner & \downarrow{\scriptstyle G} \\ \mathcal{X} & \xrightarrow[H]{} & \mathcal{Z} \end{array}$$

*If H **creates** limits of shape $\mathcal{J}$ and G **preserves** them,*
*then H' also **creates** limits of shape $\mathcal{J}$.*

Proposition (Riehl 2016, Prop. 3.3.8)

- *The functor **forget** : $\mathrm{Alg}(F) \to \mathcal{K}$ **creates** limits.*
- *The functor **dom** : $\mathcal{K}_{/O} \to \mathcal{K}$ **creates** colimits and connected limits.*

Proposition (Borceux 1994, Vol. 2, Prop. 4.3.2)

- *Since F is a left adjoint, **forget** : $\mathrm{Alg}(F) \to \mathcal{K}$ **creates** colimits.*

Completeness of categories of automata

Theorem

- *Let $\mathcal{K}$ admit **colimits of shape** $\mathcal{J}$.*
  Then $F\text{–}\mathrm{Moore}(O)$ and $F\text{–}\mathrm{Mealy}(O)$ also admit them,
  and they are computed as in $\mathcal{K}$.
- *Let $\mathcal{K}$ admit **connected limits**.*
  Then $F\text{–}\mathrm{Moore}(O)$ and $F\text{–}\mathrm{Mealy}(O)$ also admit them,
  and they are computed as in $\mathcal{K}$.

Proof. *Immediate using the characterizations of F-Mealy/F-Moore.*

Theorem

- *Let $\mathcal{K}$ admit countable products and pullbacks.*
  Then $F\text{–}\mathrm{Moore}(O)$ and $F\text{–}\mathrm{Mealy}(O)$ admit products of any finite cardinality (in particular, a terminal object),
  *but they are **not** computed as in $\mathcal{K}$.*

$\rightarrow$ *We must define discrete limits explicitly! (Terminal and products)*

Behaviour extension

- If $(\mathcal{K}, \otimes, 1)$ has countable coproducts preserved by each $- \otimes I$, a Moore automata can be extended to a span:

$$E \xleftarrow{d^*} E \otimes I^* \xrightarrow{s^*} O,$$

where $I^* := \sum_{n \geq 0} I^n$ is the freely generated monoid from I, and the morphisms d^*, s^* are defined inductively from components

$$d_n, s_n : E \times I^n \to E, O \qquad \text{for } n \geq 0.$$

Similarly, Mealy automata can be extended to $I^+ := \sum_{n \geq 1} I^n$ as

$$E \xleftarrow{d^+} E \otimes I^+ \xrightarrow{s^+} O.$$

- Intuition: extend the automata to act on *strings of symbols* instead of single inputs.

Behaviour extension of a F-automata

- An F-Moore automata $\langle E, d, s\rangle$ can be similarly extended; given

$$E \xleftarrow{d} FE \;\; ; \;\; E \xrightarrow{s} O$$

  we define the family of morphisms $s_n : F^n E \to O$ for $n \geq 0$ as the composites

$$\begin{cases} s_0 & = E \xrightarrow{s} O \\ s_1 & = FE \xrightarrow{d} E \xrightarrow{s} O \\ s_2 & = FFE \xrightarrow{Fd} FE \xrightarrow{d} E \xrightarrow{s} O \\ s_n & = F^n E \xrightarrow{F^{n-1}d} F^{n-1}E \to \cdots \xrightarrow{FFd} FFE \xrightarrow{Fd} FE \xrightarrow{d} E \xrightarrow{s} O \end{cases}$$

- In our assumption where $F \dashv R$, each map is equivalent to its mate

$$\frac{s_n : F^n E \to O}{\overline{s}_n : E \to R^n O} \text{ for } n \geq 0$$

  obtained by iterating the adjunction structure.

Skip and behaviour maps

- Each morphism obtained like this

$$\bar{s}_n : E \to R^n O$$

  is called the n-th *skip map*, since it gives the dynamics of a state after skipping n input steps.
- In case $\mathcal{K}$ has countable products, the family of all n-th skip maps $(s_n \mid n \in \mathbb{N}_{\geq 0})$ is equivalent to a single map

$$\mathrm{beh}_{\mathsf{E}} : E \to \prod_{n \geq 0} R^n O$$

  called the *behaviour map of the automata* $\mathsf{E} := \langle E, d, s \rangle$.

Terminal object

- The behaviour map has a specific universal property:

Theorem (Terminal object of F-Moore)

*The category F-Moore has a **terminal object***

$$\mathfrak{o} = \langle O_\infty, s_\infty, d_\infty \rangle, \text{ where } O_\infty = \prod_{n \geq 0} R^n O.$$

Explicitly, for any other F-Moore automata $\mathsf{E} := \langle E, d, s \rangle$, the behaviour map $\mathrm{beh}_{\mathsf{E}} : E \to O_\infty$ is the unique morphism making the following diagrams commute:

$$\begin{array}{ccc} E & \xleftarrow{\;d\;} & FE \\ {\scriptstyle \mathrm{beh}_{\mathsf{E}}}\downarrow & & \downarrow{\scriptstyle F\mathrm{beh}_{\mathsf{E}}} \\ O_\infty & \xleftarrow[d_\infty]{} & FO_\infty \end{array} \qquad \begin{array}{ccc} E & \xrightarrow{\;s\;} & O \\ {\scriptstyle \mathrm{beh}_{\mathsf{E}}}\downarrow & & \| \\ O_\infty & \xrightarrow[s_\infty]{} & O \end{array}$$

Terminal object, explicitly

- The terminal object O_∞ in a category of machines tends to be "big", since it can be obtained by Adámek's theorem as the terminal coalgebra for the functor

$$A \mapsto O \times RA \qquad \text{for } F\text{–Moore}(O),$$
$$A \mapsto RO \times RA \qquad \text{for } F\text{–Mealy}(O).$$

- The morphism d_∞ is defined using the universal property of the product by combining the family $(d_i \mid i \geq 0)$, given as

$$d_i := \frac{\prod_{n\geq 0} R^n O \xrightarrow{\pi_{i+1}} R^{i+1} O}{F(\prod_{n\geq 0} R^n O) \xrightarrow{\bar{\pi}_{i+1}} R^i O}, \quad d_\infty : F(\prod_{n\geq 0} R^n O) \to \prod_{n\geq 0} R^n O$$

and s_∞ is simply the first projection:

$$s_\infty := \prod_{n\geq 0} R^n O \xrightarrow{\pi_0} O$$

- Intuition: d_∞ advances the behaviour by one step, and s_∞ outputs.

Products in F-Moore and F-Mealy

Theorem (Products of F-automata)

Given F–Moore automata $\mathsf{E} := \langle E, d, s\rangle, \mathsf{T} := \langle T, d', s'\rangle$*, the pullback*

$$\begin{array}{ccc} P_\infty & \longrightarrow & T \\ \big\downarrow & \lrcorner & \big\downarrow \mathrm{beh}_{\mathsf{T}} \\ E & \underset{\mathrm{beh}_{\mathsf{E}}}{\longrightarrow} & O_\infty \end{array}$$

exhibits the carrier of an F–Moore automata $\mathfrak{p} := \langle P_\infty, d_P, s_P\rangle$ *that has the universal property of the* ***product*** *of* E *and* T *in* $F\text{–}\mathrm{Moore}(O)$.

- Intuition: P_∞ is the set of pairs of states $(\alpha, \beta) \in E \times T$ such that for every string of inputs $\mathrm{beh}_{\mathsf{E}}(\alpha) = \mathrm{beh}_{\mathsf{T}}(\beta)$, i.e., their behaviour coincides: P_∞ corresponds to a *bisimulation* object.

Adjoints to behaviour functors

- Our approach generalizes the one of Naudé:

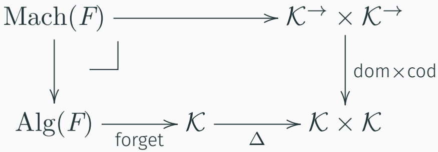
Definition

Call an endofunctor $F : \mathcal{K} \to \mathcal{K}$ an *input process* if the forgetful functor $U : \mathrm{Alg}(F) \to \mathcal{K}$ has a left adjoint G; in simple terms, an input process allows to define free F-algebras.

- Naudé [1977, 1979] concentrates on building an adjunction between a category of machines and a category of their *behaviours*

$$L : \mathrm{Beh}(F) \underset{\longleftarrow}{\overset{\longrightarrow}{\perp}} \mathrm{Mach}(F) : E$$

where $\mathrm{Mach}(F)$ is the category obtained from the pullback



and $\mathrm{Beh}(F)$ is a certain comma category on G.

Adjoints to behaviour functors

- This theorem is conceptual enough to carry over to any category of automata that can be presented as strict 2-pullback in $\mathbb{C}\mathrm{at}$ of sufficiently well-behaved functors.

Theorem

There exist functors B and L, as follows:

$$B : \mathrm{Alg}(F)_{/\langle O_\infty, d_\infty \rangle} \underset{\longleftarrow}{\overset{\longrightarrow}{\bot}} F\text{–}\mathrm{Moore}(O) : L$$

where $\langle O_\infty, d_\infty \rangle$ is the terminal (behaviour) F-algebra given.

Theorem

This is part of a longer chain of adjoints obtained as follows:

$$\mathcal{K}_{/O_\infty} \underset{\tilde{U}}{\overset{\tilde{G}}{\underset{\longleftarrow}{\overset{\longrightarrow}{\bot}}}} \mathrm{Alg}(F)_{/(O_\infty, d_\infty)} \underset{B}{\overset{L}{\underset{\longleftarrow}{\overset{\longrightarrow}{\bot}}}} F\text{–}\mathrm{Moore}(O),$$

where we denote with $\tilde{G} : \mathcal{K}_{/UA} \leftrightarrows \mathcal{H}_{/A} : \tilde{U}$ the "local" adjunction obtained from $G : \mathcal{K} \leftrightarrows \mathcal{H} : U$, with $\tilde{U}(FA, f : FA \to A) = Uf$.

Agda formalization

- We have formalized the more technical parts of our work in Agda, a dependently typed programming language and proof assistant.
- Formalization work:
  - Characterization of $F\text{–Moore}(O)\,/\,F\text{–Mealy}(O)$ as pullbacks in Cat.
  - Products and terminal objects in $F\text{–Moore}(O)$, explicitly.
  - Adjoints to behaviour functors, generalizing Naudé's approach.
  - $\mathrm{Mealy}(I, O)$ are the hom-categories of the bicategory **Mealy**.
- We use the ***agda-categories*** library as foundation to capture the basic notions of category theory.
- *(Almost 2000 lines of code!)*
- Formalization is freely available online:

https://github.com/iwilare/categorical-automata

Conclusion and Future work

- Characterizing categories of structures as *composition of simpler categories* can be a useful technique to compute limits.
- Bigger picture: the technology of category-theoretic approaches is rapidly shifting towards 2-dimensional categories as foundations for complex systems [Spivak et al. 2019], [Myers, 2021]
- Generalize other aspects of automata theory from the point of view of higher category theory (e.g. Krohn-Rhodes theorem).
- Formalizing these results in a proof assistant might pave the way for more concrete applications, where proofs act as programs to *produce and convert automata* in a provably correct way.

Thank you!