Modal Logics in **Alg**(F) and **CoAlg**(F)

By: Kevin Smith

California State University, Long Beach
Math 697 - Spring, 2020

Contents:

Section 1: Introduction

Section 2: Modal Logics
> **Languages and Formulas**
> **Frames, Models, and Satisfaction of Formulas**
> **Axioms, Rules of Inference and Deductive Systems**

Section 3: F-Algebras and F-Coalgebras

Section 4: Algebraizing Modal Logic

Section 5: Co-algebraizing Modal Logic

References

Section 1: Introduction

In this project we explore two categories for which Modal Logic can be "embedded": Algebras and Coalgebras. Each side has its own perspective on the subject and hence is able to "express" different things. We seek here to build a trifecta of contexts for our Modal Logic toolkit. In the construction, the translations between the nodes will become (at least somewhat) more apparent.

Since Modal Logic is in fact the focal point, we dedicate an entire section to fully describing what Modal Logics actually are. In Section 3, we start completely fresh from a categorical perspective and try to set up the framework for placing in the correspondents of the modal logics. In the remaining two sections, these correspondents are described.

We start in the case of general modal logics, but reduce to the case of (normal) modal logics, since there is a completeness result in this case (see pg.261 [4]).

- -

Section Citings (See Bibliography):

>> 2.1 (pg.9-14 of [4])
>> 2.2 (16-24 of [4])
>> 2.3 (31-33 and 189 of [4])
>> 3 ([5],[6],[3],[2])
>> 4 (p.262-283 and p.497-503 of [4])
>> 5

Section 2: Modal Logic

⟨⟨ 2.1 Languages and Formulas ⟩⟩

• <u>Def:</u> A **modal language** $\boldsymbol{ML(\tau,\Phi)}$ consists of a propositional language built from an **alphabet** $\boldsymbol{\Phi = \{p, q, r, ...\}}$ and a set of **connectives** $\boldsymbol{\{\neg, \vee, \wedge, \rightarrow, ...\}}$, together with a set of **modal operators** $\boldsymbol{\mathscr{O} = \{\triangle\}_{i\in I}}$ of respective **arities** $\boldsymbol{\rho = \{\rho(\triangle_i)\}_{i\in I}}$. We refer to the **modal similarity type** of $ML(\tau,\Phi)$ as the pair: $\boldsymbol{\tau = (\mathscr{O}, \rho)}$. We can summarize in a more intuitive list:

$$\boldsymbol{ML(\tau,\Phi) = \{(,), p, q, r, ..., \neg, \vee, \wedge, \rightarrow, ..., \triangle_i, ...\}}$$

• <u>Def:</u> The **grammar** or **syntax** for $ML(\tau,\Phi)$ is given by that of the propositional language, together with the recursively defined action of the modalities. Summarized as:

$$\boldsymbol{\varphi ::= p \mid \bot \mid \neg\varphi \mid \varphi_1 \vee \varphi_2 \mid \triangle_i(\varphi_1, ..., \varphi_{\rho(\triangle_i)})}$$

the set of formulas φ formed by these rules is denoted by $\boldsymbol{Form(\tau,\Phi)}$.

<u>Ex:</u> $\varphi :=$ "$p \rightarrow \Box\Diamond p$" (Canonical for Symmetry), here $\Box$ and $\Diamond$ are $\triangle$'s.

<u>Technical Notes:</u>
(1) For coding purposes, the subscripts on the modalities and formulas should be taken as in the *meta-language*, distinct identifiers should be used instead.
(2) Also *quantifiers* and *variables* used in the sequel are in the meta-language to define satisfaction etc. (although there does exist what's called a *Standard Translation* into First-Order Logic, the two should be taken to be distinct at first).

<u>Examples of Modal Languages:</u> (Specified by Their Operators)
1.) The Basic Modal Language $\boldsymbol{\{\Diamond, \Box\}}$ (both unary),
2.) Basic Temporal Language $\boldsymbol{\{\langle F\rangle, [G], \langle P\rangle, [H]\}}$ (all unary),
3.) Propositional Dynamical Logic (PDL) Language $\boldsymbol{\{\langle\pi_1\rangle, [\pi_1], \langle\pi_2\rangle, [\pi_2], ...\}}$ (all unary),
4.) Arrow Language $\boldsymbol{\{1', \otimes, \circ\}}$, arities $\{0, 1, 2\}$,

[See (p.9-14) [4]].

We purposefully ignore what a modal operator is until our discussion of satisfaction of formulas. From there it should become clearer.

⟨⟨ 2.2 Frames, Models, and Satisfaction of Formulas ⟩⟩

We have a language to work with, now we want to *instantiate* the language with models, so that we can talk about "truth".

• Def: Given a modal similarity type $\boldsymbol{\tau}$, a **$\boldsymbol{\tau}$-frame** is a pair:

$$\boldsymbol{\mathcal{F}_\tau \equiv \mathcal{F} := (W, \{R_\triangle\}_{\triangle\in\tau})}$$

where $\boldsymbol{W} \neq \emptyset$ is called the **universe** (a.k.a. *underlying set* or *state space* or *set of worlds*) and each $\boldsymbol{R_\triangle} \subseteq \underbrace{\boldsymbol{W \times ... \times W}}_{\boldsymbol{n=\rho(\triangle)+1}}$ is a **$\boldsymbol{n}$-ary relation** corresponding to $\triangle \in \tau$. Note that frames are also called *relational structures.*

• Def: Given a modal language $ML(\tau, \Phi)$, a **valuation** is a function assigning propositional letters to subsets of the universe (where they are true):

$$\boldsymbol{V : \Phi \to \mathscr{P}(W); \quad p \mapsto V(p) \subseteq W}.$$

We extend valuations to be on $Form(\Phi, \tau)$, but this requires us to define satisfaction of the modalities first.

• Def: A **$\boldsymbol{\tau}$-model** is a τ-frame together with a choice of valuation, denoted:

$$\boldsymbol{\mathfrak{M} := \big(\mathcal{F}, V\big) \equiv \Big(W, \{R_\triangle\}_{\triangle\in\tau}, V : \Phi \to \mathscr{P}(W)\Big)}$$

Hence for a given frame, there is a different model given by each choice of valuation.

• Def: Given a τ-model $\mathfrak{M}$, a state $w \in W$, and $p \in \Phi$, we say
$\boldsymbol{p}$ is satisfied at $\boldsymbol{w \in W}$ iff the state w is in the image of p under the valuation. We indicate this relationship symbolically as:

$$\boldsymbol{\Big(\mathfrak{M}, w \;\Vdash\; p\Big) \leftrightarrow \Big(w \in V(p)\Big)}$$

We also say "the model satisfies p at w".

(Continues)

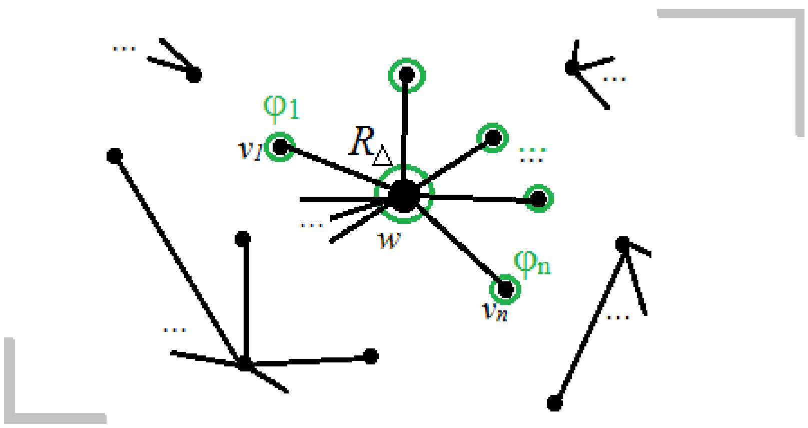
Def (Continued):

We define recursively, satisfaction of $\varphi \in \boldsymbol{Form(\tau, \Phi)}$ at $\boldsymbol{w} \in \boldsymbol{W}$

$$\text{(i)} \left(\mathfrak{M}, w \ \Vdash \ \bot\right) \text{ never (for consistency),}$$
$$\text{(ii)} \left(\mathfrak{M}, w \ \Vdash \ \neg\varphi\right) \leftrightarrow \left(w \notin V(p)\right),$$
$$\text{(iii)} \left(\mathfrak{M}, w \ \Vdash \ \varphi \vee \psi\right) \leftrightarrow \left(w \in V(\varphi) \cup V(\psi)\right),$$

and for the **modal operators**:

$$\text{(iv)} \left(\mathfrak{M}, w \ \Vdash \ \triangle(\varphi_1, ..., \varphi_n)\right) \leftrightarrow$$
$$\left(\exists v_1, ..., v_n \in W \text{ such that } R_\triangle w v_1 ... v_n \text{ and } \forall i \in \{1, ..., n\}, v_i \in V(\varphi_i)\right).$$



(Fig: Satisfaction of subformulas at some $\boldsymbol{R_\triangle}$***-accessible*** *worlds)*

Okay, so we've defined satisfaction of arbitrary formulas at a point in a model. This can be extended from a *single model* to satisfaction at the point in a *class of models* M, or at a point in a *frame* $\mathcal{F}$, or in a *class of frames* F (by applying the appropriate quantifiers). As well, each of these can be extended globally (that is, at every point in W).

• Def: We denote satisfaction in each case via "$\boldsymbol{S, w} \ \Vdash \ \varphi$" or "$\boldsymbol{S} \ \Vdash \ \varphi$" for whatever structure S we are dealing with. In particular, when S is at the level of frames (hence independent of model valuations), we say $\boldsymbol{\varphi}$ **is valid at a state** $\boldsymbol{w} \in \boldsymbol{W}$ if it is satisfied at a state in the frame or class of frames.

|<<|

Let's now talk about *consequences of true formulas.*

• <u>Def:</u> Fix a type τ. Then for a set of formulas $\mathbf{\Sigma}$, a formula φ, and a class of structures S, we say **$\boldsymbol{\varphi}$ is a local semantic consequence of $\mathbf{\Sigma}$ over $\boldsymbol{S}$** (denoted $\mathbf{\Sigma} \Vdash_{\boldsymbol{S}} \boldsymbol{\varphi}$) iff:

$$\forall \mathfrak{A} \in \boldsymbol{S}, \forall \boldsymbol{w} \in \mathfrak{A}\Big(\mathfrak{A}, \boldsymbol{w} \ \Vdash \ \mathbf{\Sigma} \implies \mathfrak{A}, \boldsymbol{w} \ \Vdash \ \varphi\Big).$$

This can of course be upgraded to a *global* definition on different structures S. We'll omit this.

Lastly,

• <u>Def:</u> The set of all formulas valid on a class of frames is denoted by $\mathbf{\Lambda}_{\mathcal{F}}$ and is referred to as the **logic for $\boldsymbol{\mathcal{F}}$**. This leads to a notion of generating sets of formulas for logics. More to come.

Before we move on to the next subsection, we should reflect a little. We just defined what it means for an abstract modal-formula to be "true", when evaluated in various classes of structures at single points or at all points. Moreover, we just defined what it means in each of the structures for a set of formulas to imply another one. This notion has a *syntactic* cousin, namely a *proof* in a deductive system.

Reference (p.33 [4]) for more on the relationship between the two cousins [Soundness and Completeness].

⟨⟨ 2.3 Axioms, Rules of Inference and Deductive Systems ⟩⟩

As mentioned in the last subsection, there is another way to go about building logics and it is abstracted from the whole frame/model/structure discussion (but intimately related).

• Def: A **modal logic Λ** is a set of modal formulas that contains all *propositional tautologies* and is closed under *modus ponens* (that is, if $\varphi \in \Lambda$ and $(\varphi \to \psi) \in \Lambda$, then $\psi \in \Lambda$) and *uniform substitution* (that is, if $\varphi \in \Lambda$ then so are all of φ's substitution instances).

• Def: Modus ponens and uniform substitution are examples of **rules of inference**, i.e. operators on the set of formulas which have as input, valid formulas, and as output, valid formulas. If Λ is generated under these rules of inference from a subset of *logically independent* formulas Γ, we call Γ the **axioms** of the logic. Together, axioms and rules of inference form what are called **deductive systems**. Our modal logics constitute a class of examples of deductive systems.

• Def: We define a **syntactic consequence symbol** "$\vdash_\Lambda$" as follows: If $\Sigma \subseteq \Lambda$ and $\psi \in \Lambda$ are such that ψ is *deducible* from a sequence of rules of inference (or from propositional calculus) from the set Σ, then we write $\boldsymbol{\Sigma \vdash_\Lambda \psi}$ and say **$\boldsymbol{\Sigma}$ proves $\boldsymbol{\psi}$**. When Λ is known, we drop reference to it in the symbol.

• Def: For such a Λ, we say if $\varphi \in \Lambda$ then it is a **theorem of $\boldsymbol{\Lambda}$** and write $\boldsymbol{\vdash_\Lambda \varphi}$ as a shorthand for $\Lambda \vdash_\Lambda \varphi$. More simply, $\vdash \varphi$.

We reduce our discussion for the next definition to the case of the *basic modal language* with binary operators $\{\Diamond, \Box\}$.

• Def: A modal logic is called **normal** if it contains the two axioms:

$$\textbf{(K)} \quad \Box(p \to q) \to (\Box p \to \Box q),$$
$$\textbf{(Dual)} \quad \Diamond p \leftrightarrow \neg\Box\neg p,$$

and is additionally closed under the rule of inference *generalization* (that is, if $\vdash \varphi$, then $\vdash \Box\varphi$). Normal modal logics generated from these and additional formulas Γ are denoted $\Lambda = \boldsymbol{K}\Gamma$. See (pg.193 [4]) for examples of such $\boldsymbol{K}\Gamma$.

This concludes the discussion for the other perspective.

Clear $\mathfrak{A}$.

Section 3: F-Algbras and F-Coalgebras

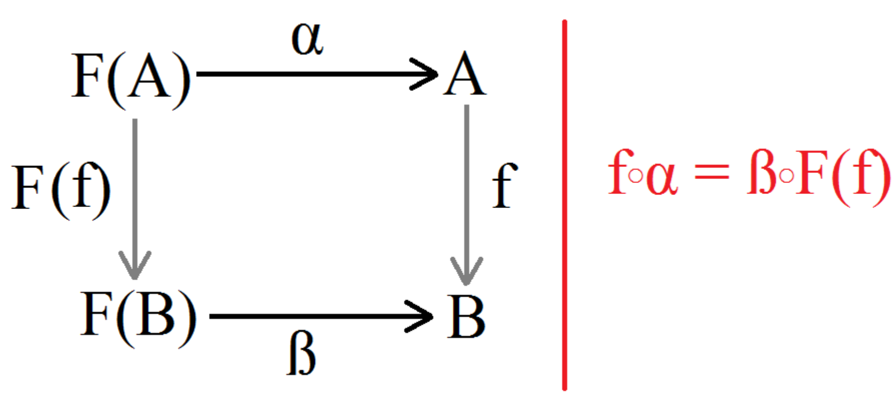
• Def: (Pg.4 [3]) An **algebra** or **algebraic structure** is a triple $\boldsymbol{\mathfrak{A} = \{A, \{f_i\}_{i \in I}, \tau\}}$, where A is the **carrier set** or *underlying set*, the f_i's are n_i-ary **operations** on A, and the **type** τ is a set containing the arities corresponding to each f_i. Moreover, we usually impose **identities** or *compatibility conditions* between the operations. (Think groups, rings, fields, vector spaces, modules, etc.)

Now recall basic categorical definitions: *category, object, morphism, dual category, and functor* as in [2]. We can generalize algebraic structures to the categorical realm in the following way based off of [5] (see alternatively in [2] (p.220)):

• Def: Taking any category $\mathcal{C}$ with $\mathfrak{A}$ as an object, with finite products, and a terminal object, the identities in an algebraic structure can be rewritten in terms of *morphisms* and *commutative diagrams* that they satisfy and these morphisms can be glued together via taking the *morphism coproduct*. What we get is a single functor, called the **signature functor** $\boldsymbol{F_{sig}}$. This is very involved and I recommend the example given for the case of groups in [5] (Exercise).

★ Def: If $\boldsymbol{F_{sig} \equiv F : \mathcal{C} \to \mathcal{C}}$ is an endofunctor on a category $\boldsymbol{\mathcal{C}}$, then an **$\boldsymbol{F}$-algebra** is a pair $\boldsymbol{(A, \alpha)} \equiv \mathfrak{A}_F$, where $\boldsymbol{A \in Obj(\mathcal{C})}$ and $\boldsymbol{\alpha \in Hom_{\mathcal{C}}(F(A), A)}$. We call A the **carrier set** (or *underlying set*) of the algebra.

• Def: A **homomorphism of $\boldsymbol{F}$-algebras** $\boldsymbol{(A, \alpha)}$ and $\boldsymbol{(B, \beta)}$ is a morphism between the carrier sets, $\boldsymbol{f \in Hom_{\mathcal{C}}(A, B)}$, such that the following diagram commutes:



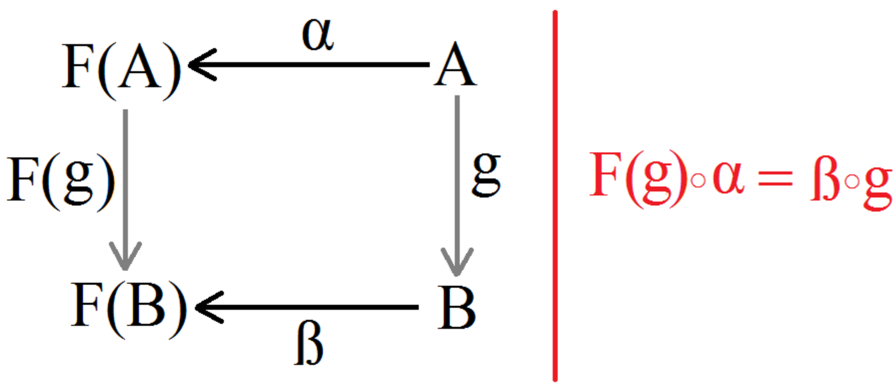
|<<|

• Def: With F-algebras as objects and the homomorphisms just defined, we get a category (as the reader should check), call this **category $\boldsymbol{Alg(F)}$**.

Referencing [6] now:

★ Def: Let $\boldsymbol{F : \mathcal{C} \to \mathcal{C}}$ be an endofunctor on a category $\boldsymbol{\mathcal{C}}$. An **$\boldsymbol{F}$-coalgebra** is a pair $(A^{op}, \alpha^{op}) \equiv \mathfrak{A}_F^{op}$, where $\boldsymbol{A \in Obj(\mathcal{C})}$ and $\boldsymbol{\alpha^{op} \in Hom_{\mathcal{C}}(A, F(A))}$. Again, A is the **carrier set**.

• Def: An **$\boldsymbol{F}$-coalgebra homomorphism** from $\boldsymbol{(A, \alpha)}$ to $\boldsymbol{(B, \beta)}$ is a morphism $\boldsymbol{g \in Hom_{\mathcal{C}}(A, B)}$ such that the following commutes:



• Def: The F-coalgebras and their associated morphisms form a **category** (as before), which we'll call $\boldsymbol{CoAlg(F)}$.

The duality is given by reversing the arrows in $\mathcal{C}$. The two categories $Alg(F)$ and $CoAlg(F)$ are not dual. (I.e. $g \neq f^{op}$ (Exercise)).

- -

Notes:
(1) The same signature functor F is used to describe all the algebras in the category $Alg(F)$ and all coalgebras in the category $CoAlg(F)$.

(2) We assume (at this point) that how ever a co-algebra is defined in the non-category-theoretic setting, it can be upgraded to this version in a way analogous to how we took algebraic structures $\mathfrak{A}$ and turned them into F-algebras (A, α) (Exercise).

Now that we have seen the beasts we are up against, let's try to observe how Blackburn et al. and Jacobs et. al. describe fitting **Modal Logic** into all of this.

(Next page)

Section 4: Algebraizing Modal Logic

Let's recall real quick that we just saw that the *duality functor* induces a *map* between the categories **Alg(F)** and **CoAlg(F)**. [Exercise: Describe this functorially].

We also saw that an element of $Alg(F)$ (i.e. an F-Algebra) is an algebraic structure $\mathfrak{A}$, phrased in a category/functor relationship $F : \mathcal{C} \to \mathcal{C}$, by $\boldsymbol{(A, \alpha : F(A) \to A)}$.

In this section, we aim to describe how **normal modal logics** [Recall (pp.6-7) **here $\Lambda_{\mathcal{F}}$**] fit into the traditional algebraic structure $\mathfrak{A}$ perspective (both semantically and syntactically), but at a very superficial level. In Section 5, we'll shoot for the (A^{op}, α^{op}) notion more directly.

- -

Paraphrasing (pp.261 and 275-283) of [4]:
On (p.261) the author states "The basic idea is to extend the algebraic treatment of classical propositional logic (which uses boolean algebras) to modal logic. The algebras employed to do this are called (Boolean Algebras with Operators (BAOs). The boolean part handles the underlying propositional logic, the additional operators handle the modalities."

On (p.275), "semantically, we deal with (an extension of) what's called a *power set algebra* that includes certain operations $m_{R_{\triangle}}$, this extension is called a *complex algebra* and it is a particular example of a BAO.

On (p.281) the syntactic side, normal modal logics are given *Lindenbaum-Tarski* algebras which are quotient algebras of the associated *formula algebras* by a certain congruence relation, together with the appropriate operations. The *Jónsson-Tarski Theorem* tells us that these Lindenbaum-Tarski algebras have *set-theoretic representations* as complex algebras."

There is clearly a lot going on here and it takes a lot of time to cover every definition involved, so we will only observe some of the main definitions mentioned in the paraphrasing. Even with these definitions, it takes a lot of work to show that these are the proper settings to model normal modal logics or logics for classes of τ-frames. We take the observers perspective here.

• Def: (p.275) Let $\tau = (\mathscr{O}, \rho)$ be a modal similarity type. A **boolean algebra with τ-operators** is an algebra:

$$\boldsymbol{\mathfrak{A} = (A, +, -, 0, f_\triangle)_{\triangle\in\tau}}$$

such that $(A, +, -, 0)$ is a **boolean algebra** and every $f_\triangle$ is an **operator** of arity $\rho(\triangle)$. That is, $f_\triangle$ is an operation satisfying:

(Normality): $f_\triangle(a_1, ..., a_{\rho(\triangle)}) = 0$ whenever $a_i = 0$ for some $i \in \{1, ..., \rho(\triangle)\}$
(Additivity): For all $i \in \{1, ..., \rho(\triangle)\}$,
$f_\triangle(a_1, ..., a_i + a_i', ..., a_{\rho(\triangle)}) = f_\triangle(a_1, ..., a_i, ..., a_{\rho(\triangle)}) + f_\triangle(a_1, ..., a_i', ..., a_{\rho(\triangle)})$

• Def: (p.267) Let A be a set. Denote the **power set of A** by **$\mathscr{P}(A)$** (the set of all subsets of A). The **power set algebra $\mathfrak{P}(A)$** is the structure:

$$\boldsymbol{\mathfrak{P}(A) = (\mathscr{P}(A), \cup, -, \emptyset),}$$

where $\emptyset$ denotes the empty set, $-$ is the operation of taking the *complement* of a set relative to A, and $\cup$ the set union operator.

Notes: "We think of A as the set of worlds and a proposition as a subset of A (the worlds that make it true). In this regard $\bot$ is $\emptyset$ and $\cup$ plays the role of $\vee$, lastly *complementation*, $-$, mirrors negation."

• Def: (p.277) For a binary relation R, (i.e. corresponding to a basic modality), on a τ-frame, we define:

$$\boldsymbol{m_R(X) = \{y \in W \mid \exists x \in X \text{ such that } R_{yx}\}}.$$

In other words, "$m_R(X)$ is the set of all states which 'see' a state in a given subset of X of the universe." (Exercise: Extend this to the general similarity types.)

• Def: (p.277) Let τ be a modal similarity type, and $\mathfrak{F} = (W, R_\triangle)_{\triangle\in\tau}$ a τ-frame. The **(full) complex algebra of $\mathfrak{F}$** (notation: $\mathfrak{F}^+$), is the expansion of the power set algebra $\mathfrak{P}(W)$ with operations $m_{R_\triangle}$ for every operator $\triangle \in \tau$. A **complex algebra** is a subalgebra of a full complex algebra. If $\boldsymbol{K}$ is a class of frames, then we denote the **class of full complex algebras of frames in $\boldsymbol{K}$** by $\boldsymbol{CmK}$.

Now that we're somewhat acquainted how modal logic is algebraized. Let's explore how it is coalgebraized.

|<<|

Section 5: Co-Algebraizing Modal Logic

We reference [1] for the remainder of this paper.

• <u>Def:</u> (p.1) Define a **coalgebra** - informally - to be a function of the form:

$$\boldsymbol{S \xrightarrow{c} [\ldots S \ldots]}$$

where $\boldsymbol{S}$ is the **state space** and $\boldsymbol{c}$ is the **transition function** or *transition structure*. The codomain $[\ldots]$ is called the **type** or **interface** of the coalgebra.

"The idea is that coalgebras describe general 'state-based systems' provided with 'dynamics' given by the function c.

It turns out there is a functor involved, as we have seen, this is phrased as:

$$\boldsymbol{(A, c : A \to F(A))}.$$

Skipping a lot of material, the author mentions the importance of so called *predicate lifting*, which we will attempt to define after the following excerpt:

• <u>Def:</u> (p.379) For a functor $\boldsymbol{F : Sets \to Sets}$ a coalgebraic modal logic is given by a *modal signature functor* $\boldsymbol{L : Sets \to Sets}$ and a *natural transformation*:

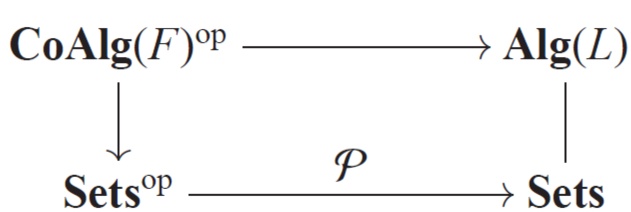
$$\boldsymbol{\delta : L\mathcal{P} \Rightarrow \mathcal{P}F},$$

where $\boldsymbol{\mathcal{P} = 2^{(-)} : Sets^{op} \to Sets}$ is the contravariant *powerset functor*.

"Given such a $\delta : L\mathcal{P} \Rightarrow \mathcal{P}F$, each F-coalgebra $c : X \to F(X)$ yields an L-algebra on the set of predicates $\mathcal{P}(X)$, namely:

$$\boldsymbol{L(\mathcal{P}(X)) \xrightarrow{\delta_X} \mathcal{P}(F(X)) \xrightarrow{c^{-1} = \mathcal{P}(c)} \mathcal{P}(X)}$$

which yields a functor $CoAlg(F)^{op} \to Alg(L)$ in a commuting diagram given by:



...Predicate lifting is described as a functor $Pred(F) : \mathcal{P} \Rightarrow \mathcal{P}F$. In the above definition this is generalised by adding a functor L in front, yielding $L\mathcal{P} \Rightarrow \mathcal{P}F$. This L makes more flexible liftings - and thus more flexible modal operators - possible."

As a final remark, Definition 1.3.2 (p.19) gives definitions for *henceforth* P and *eventually* P as two modal operators acting on *predicates* on the state space of a *sequence coalgebra*. This is a good example to look at to get a feel of where the construction is going.

There is quite a bit of overhead as far as definitions are concerned, the constructions were not as accessible as I imagined at the start of this project. There are a ton of things glossed over, but as far as the goal of understanding to some degree, the interplay between modal logic and algebraic/coalgebraic constructions, I think some parts of the battle were won. Thanks for reading!

Challenges/Exercises:

1.) Look into the creation of signature functors for F-algebras as in the discussion on (p.8)

2.) Prove $Alg(F)$ and $CoAlg(F)$ are not dual as categories (see p.9).

3.) Create signature functors from F-coalgebras (p.9).

4.) Better specify the translations in sections 4 and 5.

5.) Extend the definition of $m_R(X)$ to general similarity types.

6.) Create an example of each type of translation in the basic temporal language (for coalgebras at least see (p.22) of [1]).

References

[1] B. Jacobs: Introduction to Coalgebra, Towards Mathematics of States and Observation, Cambridge University Press (2017).

[2] H. Schubert: Categories, Springer-Verlag (1972).

[3] K. Denecke, S.L. Wismath: Universal Algebra and Applications in Theoretical Computer Science, Chapman & Hall (2002).

[4] P. Blackburn, M.D. Rijke, Y. Venema: Modal Logic, Cambridge University Press (2001).

[5] Wikipedia contributors. F-algebra. Wikipedia, The Free Encyclopedia. March 31, 2020, 05:45 UTC. Available at: https://en.wikipedia.org/w/index.php?title=F-algebra&oldid=948287509. Accessed April 10, 2020.

[6] Wikipedia contributors. F-coalgebra. Wikipedia, The Free Encyclopedia. March 18, 2020, 15:11 UTC. Available at: https://en.wikipedia.org/w/index.php?title=F-coalgebra&oldid=946171576. Accessed April 13, 2020.