

LS

The structure - semantics duality

this is the last lecture of the "Algebre" module of this course.

Until this moment we have focus on "finitary" universal algebra, sticking to operations with finite arity, as the intuition suggests

$$A^n \rightarrow A$$

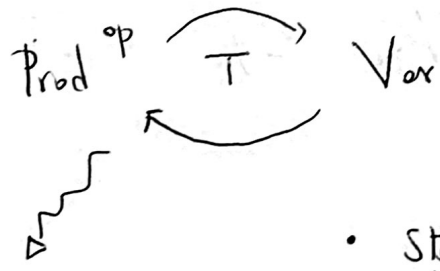
In this lecture we break the chains of "finitary algebra". Before doing so, let us recap what we did -

Syntax

Category with products finite	abstract theory
Law $Fin_S^{op} \rightarrow \mathcal{C}$ theory	Presentation of a theory
finitary monad $T: Set^S \rightarrow Set^S$	

Semantics

Variety	Semantics
variety + forgetful functor to Set^S	Structured sets
$Alg(T)$	



via this we get correspondence between

Reflections

- Starting from a finitary monad T its "theory" is $Kl(T)^{op}$ and its algebras are $Alg(T)$.

Law / Varieties equipped with a monadic functor to Set

In this lecture we "reboot" the course (up to this point) and we try to discard any finitary assumption to see how abstract nonsense makes this theory "easier".

So, let's go back where we started from: the data of a category $\mathcal{K} \xrightarrow{\mathcal{U}} \mathbf{Set}$ equipped with a forgetful functor to \mathbf{Set} .

- Unbounded implicit operations, similarly to Lawvere, we define a category whose objects are sets and ~~operations~~ morphisms are given by operations of arity unrestricted

$$\Pi_{\mathcal{U}} = \begin{cases} \rightarrow \text{obj sets } I, J \\ \rightarrow \text{morphisms } \text{Nat}(\mathcal{U}^I, \mathcal{U}^J) \end{cases}$$

of course, if \mathcal{U} has a left adjoint L , we easily get

$$\text{Nat}(\mathcal{U}^I, \mathcal{U}^J) = \text{Nat}(L(J)^{\bullet}, L(I)^{\bullet}),$$

similarly to the Grp case that we saw in the first lecture

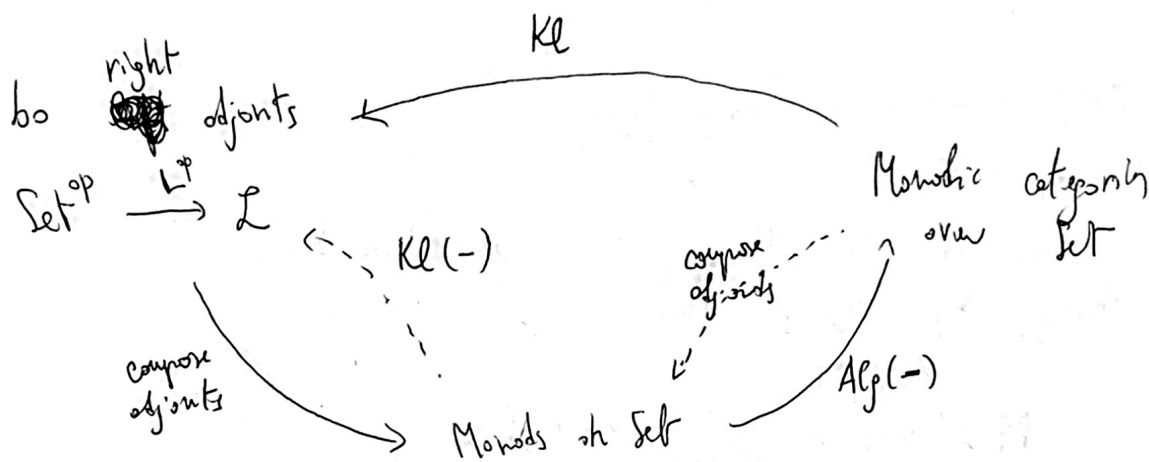
- So $\Pi_{\mathcal{U}}$ naturally comes equipped with a functor

$$\begin{array}{ccc} \mathbf{Set}^{\mathcal{P}} & \longrightarrow & \Pi_{\mathcal{U}} \\ I & \xrightarrow{\quad} & I \\ \mathbf{Set}(I, J)^{\mathcal{P}} & \xrightarrow{\quad} & \mathbf{K}(L(J), L(I)) \end{array}$$

So, given a right adjoint $K \xrightarrow{u} \text{Set}$, its "full" algebraic theory is nothing but

$$\text{Set}^{\text{op}} \xrightarrow{L^{\text{op}}} \text{Kl}(T)^{\text{op}}$$

(T is the monad associated to $L \dashv u$)



Let ... one should say that, going back to our original example $K \xrightarrow{u} \text{Set}$, our fundamental construction

$$\begin{array}{l} \text{sets} \\ \text{---} \\ \Pi_u \\ \text{---} \\ \text{Nat}(u^I, u^J) \end{array}$$

is still available, even in its "finitary" version.

And even more, we will get a comparison functor

$$K \xrightarrow{\Theta} \text{Mod}(\Pi_u)$$

↙ ↘
Set

↘ ↙
modular functor

Here models are really functors preserving any product!!

$$\Theta: K \rightarrow \mathcal{U}(K)^{(-)}$$

Here we could both choose the full algebraic Π or its finitary version

This observation is due to Linton and in the old times was called "the methodic completion" of a functor $K \rightarrow \text{Set}$ in the sense that it provides a completion making it into the best knowledge functor that approximates it.

The structure-semantic adjunction

We can rephrase the the question we have been ~~asking~~ asking by saying that

$$\text{Mod}(\text{Set})^{\text{op}} \xrightarrow{\text{Alg}(-)} \text{Cat}/\text{Set}$$

we are trying to find an adjoint for the functor $\text{Alg}(-)$, and in a sense we have provided an indirect answer given by the monad associated to Π . But there is a more direct way to answer this question.

• Codensity monad.

$$\begin{array}{ccc} K & \xrightarrow{u} & \text{Set} \\ u \downarrow & \nearrow \text{ran } u & \\ \text{Set} & & \end{array}$$

• Assume we have no problems in constructing rens. Let us show that

$\text{ran } u$ is always a monad (called codensity monad)

• $\text{ran } \mathcal{U}$ is a functor. ok.

• To provide the unit $1 \rightarrow \text{ran } \mathcal{U}$ we use the universal property or \mathcal{U} 's



with implies that to provide a map



is the same of providing a map

~~1~~ $1 \xrightarrow{\text{id}_1} \mathcal{U}$, and we choose the identity.

• to provide the multiplication $\text{ran } \mathcal{U} \circ \text{ran } \mathcal{U} \Rightarrow \text{ran } \mathcal{U}$

we observe that this is the same of providing

$$\text{ran } \mathcal{U} \circ \text{ran } \mathcal{U} \circ \mathcal{U} \Rightarrow \mathcal{U}$$

and now we use again the property $\mathcal{U}^* \dashv \text{ran}(-)$ to get

$$\text{ran } \mathcal{U} \circ \boxed{\text{ran } \mathcal{U} \circ \mathcal{U}} \xRightarrow{\text{counit}} \text{ran } \mathcal{U} \circ 1 \simeq \text{ran } \mathcal{U}$$

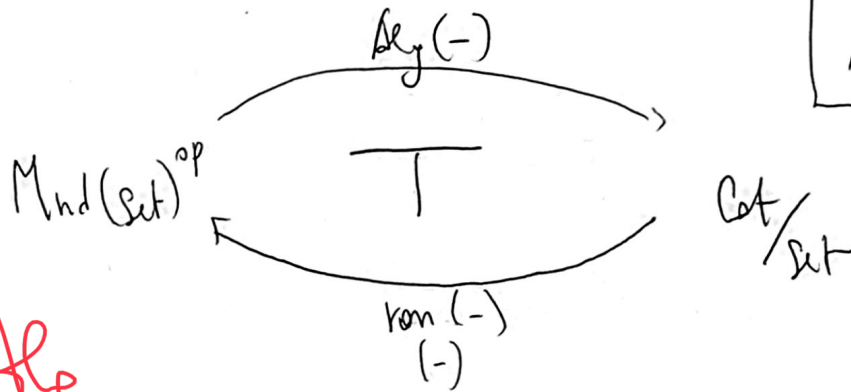
Thm (1)

$$\text{Alg}(T_u) \cong \text{Mod}(T_u)$$

the codomain model

the "full low level theory" -

Thm (2)



Structure rewrites adjunction

~~Alg~~

• ~~con(-)~~ is fully faithful!

• We checked a little bit with the existence of Kan extensions Cat/Set should be replaced with "functors for which con_u exists"

to prove thm (1) there are two strategies -

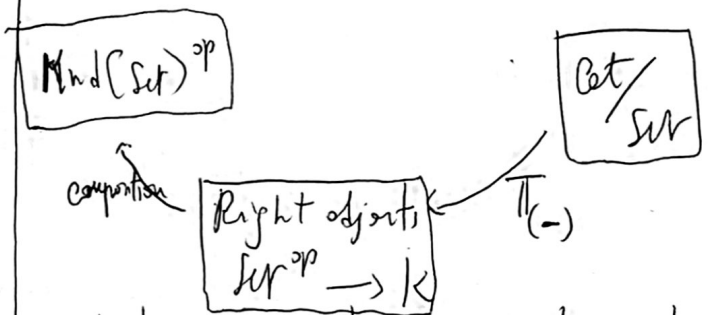
Strategy 1

Provide a functor

$$\text{Mod}(T_u) \longrightarrow \text{Alg}(T_u)$$

and show that it is an equivalence

Strategy 2



show that the composition above offers a left adjoint for $\text{Alg}(-)$ and use uniqueness of R.A.

So \prod_n^{Fin} will axiomatize all the finite model generation

- \prod_n will axiomatize ALL the natural generation

- None of these might be enough $K = \text{Cat}$

- \prod_n might be enough $K = \text{Set}$

- both might be ok $K = \text{Grp}$