Enriched Lawvere Theories Represent Operational Semantics

John C. Baez and Christian Williams

University of California, Riverside

April 12, 2018

1 Introduction

Historically, formal structures have been defined without intrinsic connection to how they actually *operate* in practical computation. In these systems, the *program* is represented separately from the *data* - but the disparity between *thought* and *action* is the source of failure. We now have a wealth of concepts which can unify the dual aspects of computation in "dynamic" structures. The idea is to *reify* operational semantics via *enrichment*.

A **Lawvere theory** [5] defines an algebraic structure abstractly, as a category $\mathcal{T}$ generated by powers of a single object S and morphisms $S^n \to S$ representing n-ary operations, satisfying equations. This represents the *theory* of a kind of algebra, which can be modelled in a category $\mathcal{C}$ by a power-preserving functor $\mu : \mathcal{T} \to \mathcal{C}$. This is a very general notion of "algebra" - computational formalisms are also presented by generators and relations: in particular, a **term calculus** represents a *formal language* by sorts, term constructors, and congruence rules.

To also represent the actual *process* of computation, we need the higher-level notion of **rewriting** one term into another. For this, ordinary categories are not enough - given two objects in the theory, the "thing" of term morphisms between them must have more structure than a set. This has been studied in the case of categories by Seely [12], posets by Ghani and Lüth [7], and others, for various related purposes. We use *edges* to represent *individual* rewrites: for this we use **graph-enriched Lawvere theories**:

sorts	: generating object S
term constructors	: generating morphisms $S^n \to S$
structural congruence	: commuting diagrams
$*$ rewrite rules	: generating hom-edges $*$

Plotkin's **operational semantics** [9] models rewriting systems with directed graphs, and this is the primary example; but there are many other useful enriching categories. Better yet, there are functors between them that allow the seamless *transition* between different kinds of operational semantics. There is a spectrum of enriching categories which forms a gradient of resolution for the semantics of term calculi. For an enriching category $\mathcal{V}$, a $\mathcal{V}$-theory is a $\mathcal{V}$-enriched Lawvere theory:

Graphs Gph-theories represent "small-step" operational semantics
- a hom-graph edge represents a *single* term rewrite.
Categories: Cat-theories represent "big-step" operational semantics:
- identity and composition represent the *reflexive-transitive* closure of the rewrite relation.
Posets: Pos-theories represent "full-step" operational semantics:
- a hom-poset boolean represents the *existence* of a big-step rewrite.
Sets: Set-theories represent denotational semantics:
- a hom-set element represents an *equivalence class* of the symmetric closure of the big-step relation.

Operational semantics is modelled and unified by enriched Lawvere theories and canonical functors between the enriching categories. This provides a more systematic categorical representation of computation.

2 Lawvere Theories

The "theory of monoids" can be defined without any reference to sets:

$$\begin{array}{rl} \text{an object} & M \\ \text{an identity element} & e : 1 \to M \\ \text{and multiplication} & m : M^2 \to M \\ \text{with associativity} & m \circ (m \times M) = m \circ (M \times m) \\ \text{and unitality} & e \circ M = M = M \circ e \end{array}$$

Lawvere theories formalize this idea. They were originally called a *finite product* theories: the category of finite sets $\mathbb{N}$ is the free category with finite coproducts on 1 - every finite set is (isomorphic to) the disjoint union of copies of $\{*\}$, and the opposite fact is that $\mathbb{N}^{\text{op}}$ is the free category with finite products on 1; so for a category with finite products $\mathcal{T}$, a strictly product-preserving bijective-on-objects functor $\iota : \mathbb{N}^{\text{op}} \to \mathcal{T}$ is essentially a category generated by one object $\iota(1) = M$ and n-ary operations $M^n \to M$, as well as the projection and diagonal morphisms of finite products.

The abstraction of this definition is powerful: the syntax encapsulates the algebraic theory, *independent* of semantics, and then one is free to realize M as almost any mathematical object. For another category with finite products $\mathcal{C}$, a **model** of the Lawvere theory in $\mathcal{C}$ is a product-preserving functor $\mu : \mathcal{T} \to \mathcal{C}$. By the "free" property above, this functor is determined by $\mu(\iota(1)) = \mu(M) = X \in \mathcal{C}$. The general theory can be thereby modelled in many useful ways. For example, ordinary groups are models $\mathcal{T}_{\text{Grp}} \to \text{Set}$, but the theory can also be modelled in the category of topological spaces to form topological groups. The models of $\mathcal{T}$ in $\mathcal{C}$ form a category $[\mathcal{T}, \mathcal{C}]_{fp}$, in which the morphisms are natural transformations.

Lawvere theories and finitary monads provide complementary representations of algebraic structures and computation [2], and they have been proven equivalent [6]. For a Lawvere theory, there is an adjunction between the category of models and the modelling category, which forms a monad on the latter; for a finitary monad, the embedding into its Kleisli category is a Lawvere theory; and these two correspondences form an equivalence of categories. The former direction is briefly reviewed to clarify the generalization to enrichment. One motivation is to use monads in an algorithm to generate sound type systems [13].

Let $\iota : \mathbb{N}^{\text{op}} \to \mathcal{T}$ be a Lawvere theory and $\text{Mod} = [\mathcal{T}, \text{Set}]_{fp}$ be the category of models. There is the **underlying set** functor $U : \text{Mod} \to \text{Set}$ which sends each model $\mu : \mathcal{T} \to \text{Set}$ to the image of the generating object, $\mu(\iota(1)) = X$ in Set. There is the **free model** functor $F : \text{Set} \to \text{Mod}$ which sends each finite set n to the representable $\mathcal{T}(n, -) : \mathcal{T} \to \text{Set}$, and in general a set X to the set of all n-ary operations on it: $\{f(x_1, ..., x_n) | f \in \mathcal{T}(n, 1), x_i \in X\}$ - this is the *filtered colimit* of representables indexed by the poset of finite subsets of X [11], which pertains to conditions of finitude in section 3. These form the adjunction:

$$\text{Mod}(F(n), \mu) \cong \mu(n) \cong \text{Set}(n, U(\mu))$$

The left isomorphism is by the Yoneda lemma, and the right isomorphism is by the definition of power in Set. Essentially, these are opposite ways of representing the n-ary operations of a given model. This adjunction induces a monad T on Set, which sends each set X to the set of all terms in the theory on X up to equality - the integral symbol is a *coend*, essentially a coproduct quotiented by terms which are equal in the theory:

$$T(X) = \int^{n \in \mathbb{N}} \text{Th}(n, 1) \times X^n$$

Conversely, for a monad T on Set, its *Kleisli category* is the category of all *free algebras* of the monad. There is a "comparison" functor $\text{Set} \to \mathcal{K}(T)$ which is the identity on objects and preserves products, so restricting to the subcategory of finite sets $\mathbb{N}$ forms the canonical Lawvere theory corresponding to the monad. This restriction is what limits the equivalence to *finitary* monads. There is a good explanation of all this in Milewski's categorical computation blog [8]. This generalizes to arbitrary *locally finitely presentable* modelling categories $\mathcal{C}$, which we will discuss in the next section.

The correspondence of Lawvere theories and finitary monads forms an equivalence of categories $Law \cong Mnd_f$, as well as an equivalence between the *models* of a theory and the *algebras* of the corresponding monad. The aforementioned references suffice; we do not need further details.

3 Enrichment

Let $(\mathcal{V}, \otimes, I)$ be a monoidal category. [4] A **$\mathcal{V}$-category** $\mathcal{C}$ is:

$$\begin{array}{rl} \text{a collection of objects} & Obj(\mathcal{C}) \\ \text{a hom-object function} & \mathcal{C}(-,-) : Obj(\mathcal{C}) \times Obj(\mathcal{C}) \to Obj(\mathcal{V}) \\ \text{composition morphisms} & \circ_{a,b,c} : \mathcal{C}(b,c) \otimes \mathcal{C}(a,b) \to \mathcal{C}(a,c) \quad \forall a,b,c \in Obj(\mathcal{C}) \\ \text{identity elements} & i_a : I \to \mathcal{C}(a,a) \quad \forall a \in Obj(\mathcal{C}) \end{array}$$

such that composition is associative and unital. A **$\mathcal{V}$-functor** $F : \mathcal{C} \to \mathcal{D}$ is:

$$\begin{array}{rl} \text{object function} & F : Obj(\mathcal{C}) \to Obj(\mathcal{D}) \\ \text{hom-functions} & F_{ab} : \mathcal{C}(a,b) \to \mathcal{D}(Fa, Fb) \quad \forall a,b \in \mathcal{C} \end{array}$$

such that F is compatible with composition and identity. A **$\mathcal{V}$-natural transformation** $\alpha : F \Rightarrow G$ is:

$$\text{a family} \quad \alpha_a : I \to \mathcal{D}(Fa, Ga) \quad \forall a \in Obj(\mathcal{C})$$

such that α is "natural" in a. See [3] for reference.

Let $\mathcal{V}$ be closed symmetric monoidal, providing

$$\begin{array}{rl} \text{internal hom} & [-,-] : \mathcal{V}^{\text{op}} \otimes \mathcal{V} \to \mathcal{V} \\ \text{symmetry braiding} & \tau_{a,b} : a \otimes b \cong b \otimes a \quad \forall a,b \in Obj(\mathcal{C}) \\ \text{tensor-hom adjunction} & \mathcal{V}(a \otimes b, c) \cong \mathcal{V}(a, [b,c]) \quad \forall a,b,c \in Obj(\mathcal{V}) \end{array}$$

Then $\mathcal{V}$ is itself a $\mathcal{V}$-category, with internal hom as the hom-object function. The tensor-hom adjunction is very important: the counit $\varepsilon_{ab} : [a,b] \otimes a \to b$ is *evaluation*, in the fundamental sense. This adjunction generalizes to an *action* of $\mathcal{V}$ on any $\mathcal{V}$-category $\mathcal{C}$: for $x \in Obj(\mathcal{V})$ and $a, b \in Obj(\mathcal{C})$, the **power** of b by x and the **copower** of a by x are objects of $\mathcal{C}$ which represent the adjunction:

$$\mathcal{C}(a \odot x, b) \cong \mathcal{V}(x, \mathcal{C}(a,b)) \cong \mathcal{C}(a, x \pitchfork b)$$

and $\mathcal{C}$ is $\mathcal{V}$-powered or copowered if all powers or copowers exist.

These are the two basic forms of enriched limit and colimit, which are not especially intuitive; but they are a direct generalization of a familiar idea in the category of sets. In Set, the power is the "exponential" function set and the copower is the product. To generalize this to an action on other Set-categories, it is key to notice:

$$X \pitchfork Y = X^Y \cong \prod_{y \in Y} X$$

$$X \odot Y = X \times Y \cong \coprod_{y \in Y} X$$

Then categories are canonically Set-powered or copowered by indexed products or coproducts of copies of an object, provided that these exist. So in the definition of Lawvere theory, even though it seems to be all about products, it is actually about powers, because these constitute the *arities* of the operations. This is precisely what is generalized in the enriched form.

There are just a few more technicalities. Given a $\mathcal{V}$-category $\mathcal{C}$, one often considers the Yoneda embedding into the $\mathcal{V}$-presheaf category $[\mathcal{C}^{\text{op}}, \mathcal{V}]$, and it is important if certain subcategories are representable; generally, some properties of $\mathcal{C}$ depend on a condition of "finitude." [1] A category is **locally finitely presentable** if it is the category of models for a *sketch*, which is a generalization of Lawvere theory to finite limits, and an object is finitely presentable or **finite** if its representable functor is *finitary*, or preserves filtered colimits. A $\mathcal{V}$-category $\mathcal{C}$ is locally finitely presentable if the underlying category $\mathcal{C}_0$ is LFP, $\mathcal{C}$ has finite powers, and $x \pitchfork - : \mathcal{C}_0 \to \mathcal{C}_0$ is finitary. The details are not crucial - all categories to be considered are locally finitely presentable. Interestingly, the category of finite sets is not. Denote by $\mathcal{V}_f$ the subcategory of $\mathcal{V}$ of finite objects - in Gph, these are simply graphs with finite vertices and edges.

4 $\mathcal{V}$-theories

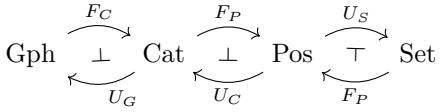
We propose a general framework in which one can *transition* seamlessly between the various forms of operational semantics. This can be done by transitioning between enriching categories by a **monoidal functor** - a functor $F : (\mathcal{V}, \otimes_\mathcal{V}, I) \to (\mathcal{W}, \otimes_\mathcal{W}, J)$ which transfers the tensor and unit via the *laxor* $\lambda : F(a) \otimes_\mathcal{W} F(b) \to F(a \otimes_\mathcal{V} b)$ and the *unitor* $\upsilon : J \to F(I)$. This induces a **change of base** functor φ (Borceux) - defining the $\mathcal{W}$-category $\widetilde{\varphi}(\mathcal{C})$:

$$\begin{array}{rl}
\text{objects} & Obj(\mathcal{C}) \\
\text{hom-function} & F \circ \mathcal{C}(-,-) : Obj(\mathcal{C}) \times Obj(\mathcal{C}) \to Obj(\mathcal{V}) \to Obj(\mathcal{W}) \\
\text{composition} & F(\circ_{a,b,c}) \circ \lambda : F(\mathcal{C}(b,c)) \otimes F(\mathcal{C}(a,b)) \to F(\mathcal{C}(b,c) \otimes \mathcal{C}(a,b)) \to F(\mathcal{C}(a,c)) \quad \forall a,b,c \in Obj(\mathcal{C}) \\
\text{identity} & F(i_a) \circ \upsilon : J \to F(I) \to F(\mathcal{C}(a,a)) \quad \forall a \in Obj(\mathcal{C})
\end{array}$$

These rather abstract notions are necessary to define the central concept: a $\mathcal{V}$-enriched Lawvere theory, or **$\mathcal{V}$-theory**, is a finitely-powered $\mathcal{V}$-category Th equipped with a strictly power-preserving bijective-on-objects $\mathcal{V}$-functor $\iota : \mathcal{V}_f^{\mathrm{op}} \to \mathrm{Th}$. A **model** of a $\mathcal{V}$-theory is a finite-power $\mathcal{V}$-functor $\mu : \mathrm{Th} \to \mathcal{V}$, and $\mathcal{V}$-natural transformations between them form the $\mathcal{V}$-category of models $FP(\mathrm{Th}, \mathcal{V})$.

This generalizes to enriched categories: $\mathcal{V}$-theories are equivalent to finitary $\mathcal{V}$-monads [10]. The basic idea is the same, but there are subtleties.

5 Semantics



References

[1] J. Adamek and J. Rosicky, *Locally presentable and accessible categories*, London Mathematical Society Lecture Notes Series **189** (1994).

[2] Martin Hyland and John Power, *The category theoretic understanding of universal algebra: Lawvere theories and monads*, Electronic Notes in Theoretical Computer Science **172** (2007).

[3] G.M. Kelly, *Basic concepts of enriched category theory*, vol. 64, 1982.

[4] Saunders Mac Lane, *Categories for the working mathematician*, Graduate Texts in Mathematics (1971).

[5] F. William Lawvere, *Functorial semantics of algebraic theories*, Reports of the Midwest Category Seminar II **5** (2004).

[6] F.E.J. Linton, *Some aspects of equational theories*, Proceedings of the Conference on Categorical Algebra (1965).

[7] Christoph Lüth and Neil Ghani, *Monads and modular term rewriting*, Lecture Notes in Computer Science (1997).

[8] Bartosz Milewski, *Lawvere theories*, https://bartoszmilewski.com/2017/08/26/lawvere-theories/, 2017.

[9] Gordon D. Plotkin, *A strucural approach to operational semantics*, Journal of Logic and Algebraic Programming (1981).

[10] John Power, *Enriched lawvere theories*, Theory and Applications of Categories (2000).

[11] Urs Schreiber, *Lawvere theory*, (2010).

[12] R.A.G. Seely, *Modelling computations: A 2-categorical framework*, Symposium on Logic in Computer Science (1987).

[13] Michael Stay and L.G. Meredith, *Logic as a distributive law*, Logic in Computer Science (2016).