Graded Algebraic Theories

Satoshi Kura

National Institute of Informatics, Tokyo, Japan

The Graduate University for Advanced Studies (SOKENDAI), Kanagawa, Japan

Outline

Algebraic theory for graded monads

- Graded algebraic theory
- Graded Lawvere theory

Application: combining effects (see my paper)

- Sum
- Tensor

Monads

- Semantics for computational effects
  [Moggi, LICS'89]
- Monads correspond to
  - algebraic theories (= operations + equations)
  - Lawvere theories [Lawvere, PhD thesis]

(finitary) monads	$\simeq$	algebraic theories	$\simeq$	Lawvere theories

Algebraic presentation

Computational effects can be algebraically presented.

- Arity = number of continuations

  $\mathbf{lookup}_l(\lambda u.x_u)$ "choose x_u by value u in l"

- Executed from outermost operation

$$\mathbf{update}_{l,v}(\mathbf{lookup}_l(\lambda u.x_u))$$

"update l with v and then lookup l"

An example of algebraic theories

State monad $S \to (-) \times S$

where $S = V^{\mathsf{Loc}}$

Algebraic theory $\mathcal{T}^{st}$:

- $\mathbf{update}_{l,v}$: unary
- $\mathbf{lookup}_l$: V-ary
- with several equational axioms
  - e.g. $\mathbf{update}_{l,v}(\mathbf{lookup}_l(\lambda u.x_u)) = x_v$

terms in $\mathcal{T}^{st}$ = stateful computations

Effect systems

- estimate the scope of computational effects
  - accessed memory locations
  - raised exceptions
- via type systems

$$\frac{\Gamma \vdash M : T(e, \tau) \qquad \Gamma, x : \tau \vdash N : T(e', \sigma)}{\Gamma \vdash \mathrm{let}\ x\ \mathrm{be}\ M\ \mathrm{in}\ N : T(e \cdot e', \sigma)}$$

- semantics by graded monads

  [Katsumata, POPL'14]

Graded monads

$\mathbb{M} = (\mathbb{M}, I, \otimes)$: monoidal category.

$\mathbb{M}$-graded monad on category $\mathcal{C}$ consists of

- a functor $T : \mathbb{M} \to [\mathcal{C}, \mathcal{C}]$
- two natural transformations

$$\eta : \mathrm{Id}_{\mathcal{C}} \to T_I$$

$$\mu_{m_1,m_2} : (T_{m_1}) \circ (T_{m_2}) \to T_{m_1 \otimes m_2}$$

- satisfying monad laws.

Algebraic theories for graded monads

Only a few pieces of work.

- When $\mathbb{M} = (\mathbb{N}, 0, +)$: [Milius, CALCO'15]

Why not consider

- more general $\mathbb{M}$?
  - "subeffecting" $m \leq m'$?
- "graded" Lawvere theories?

Extending algebraic/Lawvere theories

$$\begin{tikzcd}
\text{(finitary) monads} \arrow[r, phantom, "\simeq"] \arrow[d] & \text{algebraic theories} \arrow[r, phantom, "\simeq"] \arrow[d, "\text{generalizing [Milius, CALCO'15]}"] & \text{Lawvere theories} \arrow[d] \\
\text{(finitary) } \mathbb{M}\text{-graded monads} \arrow[r, phantom, "\simeq"] & \textcolor{orange}{\mathbb{M}\textbf{-graded}} \text{ algebraic theories} \arrow[r, phantom, "\simeq"] & \textcolor{orange}{\mathbb{M}\textbf{-graded}} \text{ Lawvere theories}
\end{tikzcd}$$

[Smirnov'08]

$\mathbb{M}$: small strict monoidal category

($\mathbb{M}$ is not necessarily symmetric)

Graded algebraic theories

Graded algebraic theories

$$\mathcal{T} = (\Sigma, E)$$

- Σ: operations (equipped with **grade**)
- E: equational axioms

Operations

Each operation has its grade $m \in \mathrm{M}$.

$$\Sigma = (\Sigma_{n,m})_{n \in \mathbb{N}, m \in \mathrm{M}}$$

Example (graded state monad).

grade: accessed memory locations

$$\mathrm{M} = ((2^{\mathsf{Loc}}, \subseteq), \emptyset, \cup)$$

operation	arity	grade
$\mathbf{lookup}_l$	V	$\{l\}$
$\mathbf{update}_{l,v}$	1	$\{l\}$

Terms

We define the set of terms with grade $m \in \mathbb{M}$

$$T_m^{\Sigma}(X)$$

where X is a set of variables.

Three rules of term formation:

- Variables: x
- Operations: $f(t_1, \ldots, t_n)$
- Coercions (or subeffecting): $c_{w:m \to m'}(t)$

Grade of Terms

$$\underbrace{x}_{I}$$

$$\underbrace{\overbrace{f}^{m}\ (\overbrace{t_1, \ldots, t_n}^{m'})}_{m \otimes m'}$$

$$\underbrace{c_{w:m \to m'}(\overbrace{\ t\ }^{m})}_{m'}$$

Equational Axioms

Grades of both sides must be equal.

$$E = \Big(E_m \subseteq T^{\Sigma}_m(X) \times T^{\Sigma}_m(X)\Big)_{m \in \mathrm{M}}$$

Example (graded state monad).

$$\underbrace{\mathbf{update}_{l,v}(\mathbf{lookup}_l(\lambda u.x_u))}_{\{l\}} = c_{\emptyset \subseteq \{l\}}(\overbrace{x_v}^{\emptyset})$$

(+ several other equations)

Equivalence

terms with grade m	$=$	computations with effect m

Theorem.

category of $\mathbb{M}$-graded algebraic theories	$\simeq$	category of finitary[1] $\mathbb{M}$-graded monads on $\mathbf{Set}$

[1] graded monad T where $T_m : \mathbf{Set} \to \mathbf{Set}$ is finitary for each $m \in \mathbb{M}$

Graded Lawvere theories

Why Lawvere theories?

Independent from specific choice of operations

- because we consider all terms generated by algebraic theories

Independent from base category

- compared to monads, which are defined on one base category

Intuition of Lawvere theories

A Lawvere theory is a category L where

- objects: arities

$$\mathrm{ob}L = \mathbb{N}$$

- arrows: terms (modulo equational axioms)

$$n \xrightarrow{(t_1,\ldots,t_k)} k$$

- composition: substitution

$$n \xrightarrow{(t_1,\ldots,t_k)} k \xrightarrow{s} 1 \quad = \quad n \xrightarrow{s[t_1/x_1,\ldots,t_k/x_k]} 1$$

Lawvere theories

Definition.

A Lawvere theory is a pair of

- a category L with finite products and
- a strict finite-product preserving identity-on-objects functor $J : \aleph_0^{\mathrm{op}} \to L$.

Here, $\aleph_0$ is the full subcategory of Set on natural numbers.

Graded Lawvere theories?

Different terms have different grades.

$$\circ \quad \begin{matrix} \xrightarrow[\text{grade } m]{t} \\ \xrightarrow[\text{grade } m']{t'} \\ \vdots \end{matrix} \quad \circ$$

Take hom-objects not from $\mathbf{Set}$

but from $[\mathbf{M}, \mathbf{Set}]$.

Graded Lawvere theories?

How to express the mathematical structure of grades?

$$n \xrightarrow[m]{(t_1,\ldots,t_k)} k \xrightarrow[m']{s} 1 \quad = \quad n \xrightarrow[m' \otimes m]{s[t_1/x_1,\ldots,t_k/x_k]} 1$$

Consider $[\mathbb{M}, \mathrm{Set}]$**-enriched** categories using the **Day tensor monoidal structure**.

$[\mathrm{M}, \mathrm{Set}]$-categories

If C is an $[\mathrm{M}, \mathrm{Set}]$-category,

- hom-object: $\quad \boldsymbol{C(X,Y)} : \mathbf{M} \to \mathbf{Set}$

$$\boldsymbol{X} \xrightarrow[\text{grade } m]{\boldsymbol{f}} \boldsymbol{Y} \quad \iff \quad \boldsymbol{f \in C(X,Y)m}$$

- composition:

$$\boldsymbol{X} \xrightarrow[m]{f} \boldsymbol{Y} \xrightarrow[m']{g} \boldsymbol{Z} \quad = \quad \boldsymbol{X} \xrightarrow[m' \otimes m]{g \circ f} \boldsymbol{Z}$$

Graded Lawvere Theories

Definition.

A graded Lawvere theory is a pair of

- an $[\mathbb{M}, \mathbf{Set}]$-category $\boldsymbol{L}$ with $\boldsymbol{N}_{\mathbb{M}}^{\mathrm{op}}$-cotensors and
- a $\boldsymbol{N}_{\mathbb{M}}^{\mathrm{op}}$-cotensor preserving identity-on-objects functor $\boldsymbol{J} : \boldsymbol{N}_{\mathbb{M}}^{\mathrm{op}} \to \boldsymbol{L}$.

$\boldsymbol{N}_{\mathbb{M}}$ is the full sub-$[\mathbb{M}, \mathbf{Set}]$-category of $[\mathbb{M}, \mathbf{Set}]$ on $\{n \cdot y(I) \mid n \in \mathbb{N}\}$ (if $\mathbb{M}$ is symmetric).

Equivalence

Theorem.

category of $\mathbb{M}$-graded Lawvere theories	$\simeq$	category of finitary[2] $\mathbb{M}$-graded monads on $\mathbf{Set}$

[2]graded monad T where $T_m : \mathbf{Set} \to \mathbf{Set}$ is finitary for each $m \in \mathbb{M}$

Conclusions

- Graded algebraic theories: by assigning a grade to each term.
- Graded Lawvere theories: by considering enrichment in $[\mathbf{M}, \mathbf{Set}]$.
- They correspond to graded monads.
- Sums and tensors can be extended to graded algebraic theories.

Appendix

Combining effects

For ordinary algebraic theories

[Hyland & Power, TCS'06]

Sum: add operations with no additional equation

$$\mathcal{T}_1 \qquad \mathcal{T}_2 \qquad \mapsto \qquad \mathcal{T}_1 + \mathcal{T}_2$$

Tensor: allow commutation $f(g(x)) = g(f(x))$ between two theories

$$\mathcal{T}_1 \qquad \mathcal{T}_2 \qquad \mapsto \qquad \mathcal{T}_1 \otimes \mathcal{T}_2$$

Combining effects

For graded algebraic theories:

Sum

$$\begin{array}{ccccc} \mathrm{M}\text{-graded} & & & & \mathrm{M}\text{-graded} \\ \mathcal{T}_1 & \mathcal{T}_2 & \mapsto & & \mathcal{T}_1 + \mathcal{T}_2 \end{array}$$

Tensor

$$\begin{array}{cccc} \mathrm{M}_1\text{-graded} & \mathrm{M}_2\text{-graded} & & \mathrm{M}_1 \times \mathrm{M}_2\text{-graded} \\ \mathcal{T}_1 & \mathcal{T}_2 & \mapsto & \mathcal{T}_1 \otimes \mathcal{T}_2 \end{array}$$

Sums

Given two $\mathbb{M}$-graded algebraic theories

$$\mathcal{T} = (\Sigma, E), \qquad \mathcal{T}' = (\Sigma', E')$$

where

- Σ, Σ': sets of operations
- E, E': sets of equations

The sum is

$$\mathcal{T} + \mathcal{T}' := (\Sigma \cup \Sigma', E \cup E')$$

Example of Sums

finitary monad $\boldsymbol{T} : \mathbf{Set} \to \mathbf{Set}$ $+$ graded exception monad $\boldsymbol{T}^{\mathrm{ex}} : \mathrm{M} \to [\mathbf{Set}, \mathbf{Set}]$

$$= \quad \boldsymbol{T}\boldsymbol{T}^{\mathrm{ex}} : \mathrm{M} \to [\mathbf{Set}, \mathbf{Set}]$$

Cf.

finitary monad $\boldsymbol{T}$ $+$ exception monad $(-) + \mathbf{Ex}$

$$= \quad \boldsymbol{T}((-) + \mathbf{Ex})$$

Tensors in [Hyland & Power, TCS'06]

Tensors are defined by

$$(\Sigma, E) \otimes (\Sigma', E') := (\Sigma \cup \Sigma', E \cup E' \cup E_\otimes)$$

where $E_\otimes$ consists of

$$f(\lambda i.g(\lambda j.x_{ij})) = g(\lambda j.f(\lambda i.x_{ij}))$$

for any $f \in \Sigma$ and $g \in \Sigma'$.

Problem of commutativity

$$\begin{array}{ccccc} \mathrm{M}\text{-graded} & & & \mathrm{M}\text{-graded} & \\ \mathcal{T} \qquad \mathcal{T}' & & \mapsto & \mathcal{T} \otimes \mathcal{T}' & \text{(?)} \end{array}$$

We cannot define

$$\underbrace{f}_{m}\ (\lambda i.\ \underbrace{g}_{m'}\ (\lambda j.x_{ij})) = \underbrace{g}_{m'}\ (\lambda j.\ \underbrace{f}_{m}\ (\lambda i.x_{ij}))$$

because M is not necessarily symmetric.

$$m \otimes m' \neq m' \otimes m$$

Tensors

$$\begin{array}{cccc} \mathrm{M}\text{-graded} & \mathrm{M}'\text{-graded} & & \mathrm{M}\times\mathrm{M}'\text{-graded} \\ \mathcal{T} & \mathcal{T}' & \mapsto & \mathcal{T}\otimes\mathcal{T}' \end{array}$$

Idea: embed M and M' into $\mathrm{M} \times \mathrm{M}'$ by

$$\begin{aligned} \boldsymbol{m} \mapsto (\boldsymbol{m}, \boldsymbol{I}') \quad &: \quad \mathrm{M} \to \mathrm{M} \times \mathrm{M}' \\ \boldsymbol{m}' \mapsto (\boldsymbol{I}, \boldsymbol{m}') \quad &: \quad \mathrm{M}' \to \mathrm{M} \times \mathrm{M}' \end{aligned}$$

Then, we can define

$$\underbrace{\boldsymbol{f}}_{(\boldsymbol{m},\boldsymbol{I}')}\ (\boldsymbol{\lambda i}.\ \underbrace{\boldsymbol{g}}_{(\boldsymbol{I},\boldsymbol{m}')}\ (\boldsymbol{\lambda j}.\boldsymbol{x}_{ij})) = \underbrace{\boldsymbol{g}}_{(\boldsymbol{I},\boldsymbol{m}')}\ (\boldsymbol{\lambda j}.\ \underbrace{\boldsymbol{f}}_{(\boldsymbol{m},\boldsymbol{I}')}\ (\boldsymbol{\lambda i}.\boldsymbol{x}_{ij}))$$

Example of Tensors

Example: theories for graded state monad

Suppose

- $\mathcal{T}_1$: $\mathbf{2}$-graded theory with $\#\mathbf{Loc} = 1$
- $\mathcal{T}_n$: $\mathbf{2}^n$-graded theory with $\#\mathbf{Loc} = n$
- $\mathbf{2} = \{\bot \leq \top\}$ and $\mathbf{2}^n$ are join-semilattices

We have

$$\mathcal{T}_n = \underbrace{\mathcal{T}_1 \otimes \cdots \otimes \mathcal{T}_1}_{n\text{-fold tensor}}$$