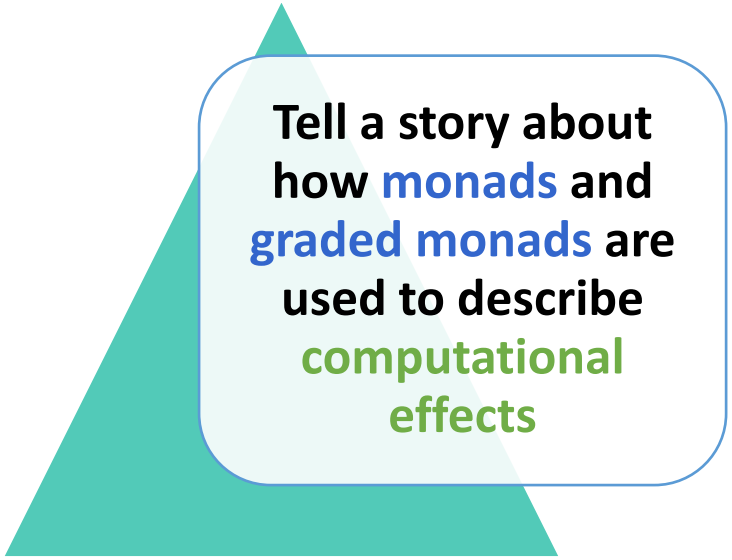
Monads, Graded Monads, and Computational Effects

CATEGORICAL LATE LUNCH

Nuiok Dicaire

31 March 2021

Objective/Outline



1. Monads and examples
2. Monads and effects
3. Graded monads and examples
4. Algebras for graded monads (formal theory of graded monads)

Perrone, Paolo. "Notes on Category Theory with examples from basic mathematics." *arXiv preprint arXiv:1912.10642* (2019).

Milewski, Bartosz. *Category theory for programmers*. Blurb, 2018.

Fujii, Soichiro, Shin-ya Katsumata, and Paul-André Mellies. "Towards a formal theory of graded monads." *International Conference on Foundations of Software Science and Computation Structures*. Springer, Berlin, Heidelberg, 2016.

What is a monad?

A way to talk about generalised elements and generalised functions in an extended space

- Let $\mathbf{C}$ be a category
- Triple: (T, μ, η)

Endofunctor

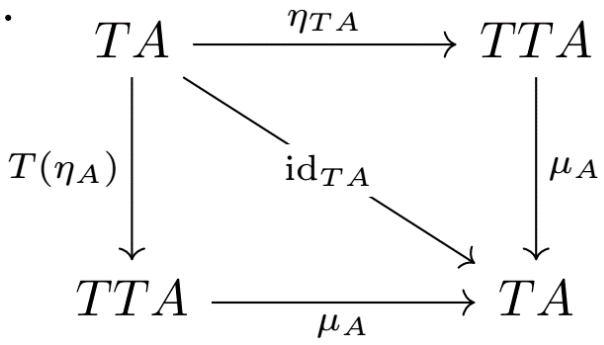
Natural Transformations

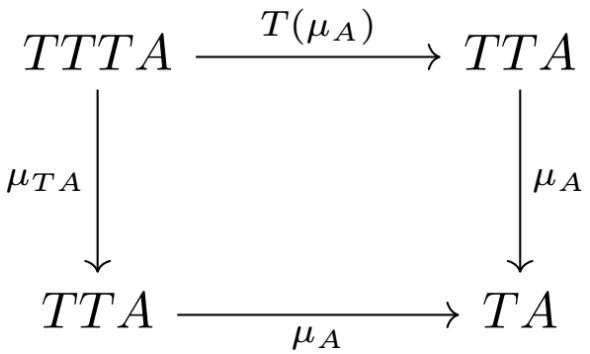
$$T : \mathbf{C} \to \mathbf{C}$$

$$\mu : TT \Rightarrow T$$

$$\eta : \mathrm{id}_{\mathbf{C}} \Rightarrow T$$

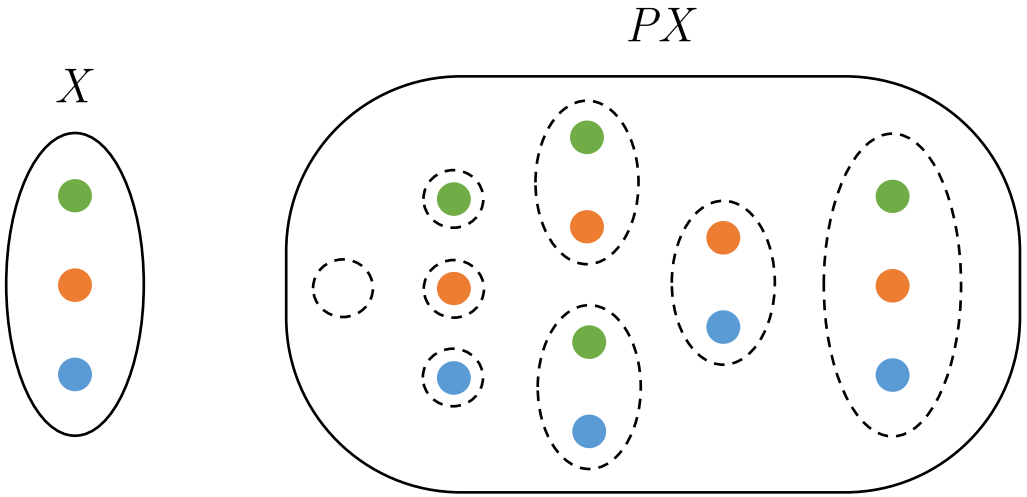
- Such that:

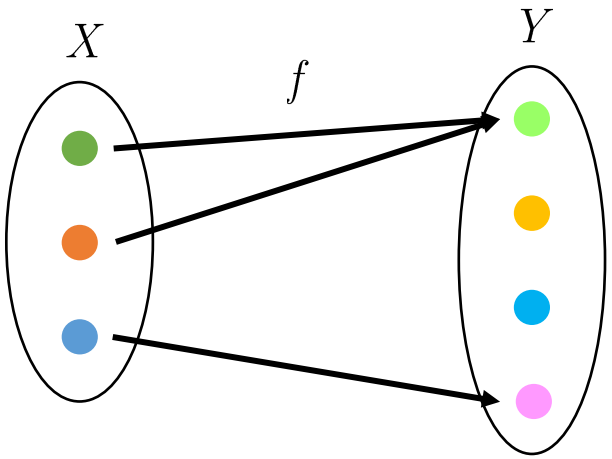




Ex: Power set monad

$$P : \mathbf{Set} \longrightarrow \mathbf{Set}$$
$$X \longrightarrow PX$$
$$f_{X \to Y} \longrightarrow P(f)$$





η: maps to singletons

μ: union

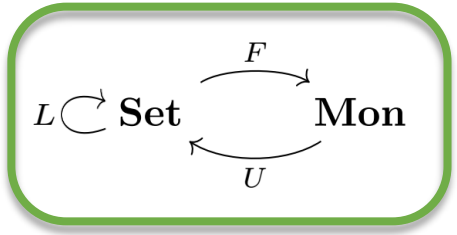
Ex: List monad

(Free monoid monad)

$L : \mathbf{Set} \longrightarrow \mathbf{Set}$

$X \longmapsto \mathrm{List(X)}$

free monoid (all finite lists)



$\eta_X : a \mapsto [a]$

μ_X : flattening

e.g. $[[a, b, c], [c], [c, b]] \mapsto [a, b, c, c, c, b]$

Ex: Maybe monad

(Option monad)

$M : \mathbf{C} \longrightarrow \mathbf{C}$

$A \longmapsto A \amalg *$

terminal object

η: inclusion

μ: maps both added $*$ in MMA to the one added $*$ in MA

Ex: Writer monad

(Action monad)

$$W : \mathbf{C} \longrightarrow \mathbf{C}$$
$$A \longmapsto M \otimes A$$

$\mathbf{C}$ monoidal category
M monoid in $\mathbf{C}$
$e : 1 \to M$
$\cdot : M \otimes M \to M$

$$\eta_A : A \xrightarrow{\cong} 1 \otimes A \xrightarrow{e \otimes \mathrm{id}} M \otimes A$$

$$\mu_A : M \otimes (M \otimes A) \xrightarrow{\cong} (M \otimes M) \otimes A \xrightarrow{\cdot \otimes \mathrm{id}} M \otimes A$$

Ex: State monad

$$T : \mathbf{C} \longrightarrow \mathbf{C}$$
$$A \longmapsto \Sigma \multimap (\Sigma \otimes A)$$
$$f_{A \to B} \longmapsto T(f)$$

$$\frac{S \multimap (S \otimes A) \xrightarrow{T(f)} S \multimap (S \otimes B)}{[S \multimap (S \otimes A)] \otimes S \xrightarrow{ev} S \otimes A \xrightarrow{\mathrm{id}_S \otimes f} S \otimes B}$$

η: curry of identity

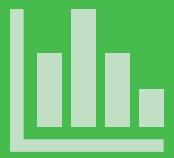
μ: curry of $ev \circ ev$

Things that programs do:

Interactive Input/Output

Side-effects (access/modify state): read or write

Exceptions or errors



Randomness

Continuation

Monads in computer science

Functional programming languages (Haskell)

- "For a given input, the output is deterministic"
- Composing the problem as a set of functions to be executed

Pure vs. effectful

- How to describe side-effects in a pure functional language?

Solution: Monads

Kleisli category: Composing "embellished" morphism

- Let (T, η, μ) be a monad on $\mathbf{C}$.

Kleisli morphism:

$$k : X \to TY$$

Writer monad on **Set**:

$$X \longrightarrow M \times Y$$

Outputs Y and a log of actions

Kleisli composition

$$k : X \to M \times Y$$
$$h : Y \to M \times Z$$

The composition $h \circ_{\mathrm{kl}} k : X \to M \times Z$ is:

$$\begin{array}{ccccccc} X & \xrightarrow{k} & M \times Y & \xrightarrow{\mathrm{id}_M \times h} & M \times M \times Z & \xrightarrow{+ \times \mathrm{id}_Z} & M \times Z \\ x & \longmapsto & (m, y) & \longmapsto & (m, n, z) & \longmapsto & (m + n, z) \end{array}$$

Concatenation of log files

This defines the Kleisli category (i.e. category of free T-algebras)

All or nothing?

Grading of effects

Ex: Permissions

Graded monads

(parametric monads)

Definition

A ***graded monad*** is a lax monoidal functor $T : (\mathcal{M}, \otimes, I) \to ([\mathbf{C}, \mathbf{C}], \circ, \mathrm{id}_{\mathbf{C}})$

monoidal category

$$(\mathcal{A}, \otimes_{\mathcal{A}}, 1_{\mathcal{A}}) \longrightarrow (\mathcal{B}, \otimes_{\mathcal{B}}, 1_{\mathcal{B}})$$

- Functor $F : \mathcal{A} \to \mathcal{B}$
- Morphism $\eta : 1_{\mathcal{B}} \to F(1_{\mathcal{A}})$
- Natural transformation
  $\mu_{a,a'} : F(a) \otimes_{\mathcal{B}} F(a') \to F(a \otimes_{\mathcal{A}} a')$
  for each a, a' in $\mathcal{A}$

1. Functor $T_m : \mathcal{C} \to \mathcal{C}$
2. $T(f) : T_m \Rightarrow T_n$
3. $\eta^A : A \Rightarrow T_I(A)$
4. $\mu^A_{m,n} : T_m T_n(A) \Rightarrow T_{m \otimes n}(A)$

Note: Monads are **1**-graded monads

Graded monads

(parametric monads)

Definition

A ***graded monad*** is a lax monoidal functor $T : (\mathcal{M}, \otimes, I) \to ([\mathbf{C}, \mathbf{C}], \circ, \mathrm{id}_{\mathbf{C}})$

monoidal category

$$\begin{array}{ccc} T_n(A) & \xrightarrow{T_n(\eta^A)} & T_nT_I(A) \\ {\scriptstyle \eta^{T_n(A)}}\downarrow & \searrow^{\mathrm{id}} & \downarrow{\scriptstyle \mu^A_{n,I}} \\ T_IT_n(A) & \xrightarrow[\mu^A_{I,n}]{} & T_n(A) \end{array} \qquad \begin{array}{ccc} T_mT_nT_p(A) & \xrightarrow{\mu^{T_p(A)}_{m,n}} & T_{m\otimes n}T_p(A) \\ {\scriptstyle T_M(\mu^A_{n,p})}\downarrow & & \downarrow{\scriptstyle \mu^A_{m\otimes n,p}} \\ T_mT_{n\otimes p}(A) & \xrightarrow[\mu^A_{m,n\otimes p}]{} & T_{m\otimes n\otimes p}(A) \end{array}$$

$T : \mathbf{C} \to \mathbf{C}$

$\mu : TT \Rightarrow T$

$\eta : \mathrm{id}_{\mathbf{C}} \Rightarrow T$

$$\begin{array}{ccc} TA & \xrightarrow{\eta_{TA}} & TTA \\ {\scriptstyle T(\eta_A)}\downarrow & \searrow^{\mathrm{id}_{TA}} & \downarrow{\scriptstyle \mu_A} \\ TTA & \xrightarrow[\mu_A]{} & TA \end{array} \qquad \begin{array}{ccc} TTTA & \xrightarrow{T(\mu_A)} & TTA \\ {\scriptstyle \mu_{TA}}\downarrow & & \downarrow{\scriptstyle \mu_A} \\ TTA & \xrightarrow[\mu_A]{} & TA \end{array}$$

Ex: Graded writer monad

Start with the monoid $\mathbb{E} := (P(\Sigma^*), \subset, \{\mathcal{E}\}, \cdot)$

where $m \cdot n = \{k@l \mid k \in m, l \in n\}$

Make $\mathbb{E}$ into a monoidal category $\mathcal{M}$.

Define $W : (\mathcal{M}, \cdot, \{\mathcal{E}\}) \longrightarrow ([\mathbf{Set}, \mathbf{Set}], \circ, \mathrm{id}_{\mathbf{Set}})$

$$m \longmapsto W_m : \mathbf{Set} \to \mathbf{Set}$$
$$A \mapsto m \times A$$

$$\eta^A : A \longrightarrow W_{\{\mathcal{E}\}}(A) = \{\mathcal{E}\} \times A$$
$$x \longmapsto (\mathcal{E}, x)$$

$$\mu^A_{m,n} : W_m W_n(A) \longrightarrow W_{m \cdot n}(A)$$
$$m \times n \times A \longrightarrow (m \cdot n) \times A$$
$$(k, l, A) \longmapsto (k@l, A)$$

Katsumata, Shin-ya. "Parametric effect monads and semantics of effect systems." *Proceedings of the 41st ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages*. 2014.

$$\frac{\mathbb{N}}{\begin{array}{l}\text{ob: } n \in \mathbb{N}\\ \text{maps: } \leq\\ \otimes\text{: } +\\ I\text{: } 0\end{array}}$$

Ex: Graded state monad

$$\eta^A : A \longrightarrow T_{S^0} = I_{\mathbf{C}} \multimap (I_{\mathbf{C}} \otimes A) \cong A$$

- $\mathbf{C}$ symmetric monoidal closed category.
- S object of $\mathbf{C}$

$$\mu^A_{m,n} : T_{S^m} T_{S^n}(A) \longrightarrow T_{S^{m+n}}(A)$$

$$S^m \multimap (S^m \otimes (S^n \multimap (S^n \otimes A))) \longrightarrow S^{m+n} \multimap (S^{m+n} \otimes A)$$

$$F : (\mathbb{N}, +, 0) \longrightarrow ([\mathbf{C}, \mathbf{C}], \circ, \mathrm{id}_{\mathbf{C}})$$

$$m \longmapsto T_{S^m} = S^m \multimap (S^m \otimes -)$$

$$m \leq n \longmapsto T_{S^m} \Rightarrow T_{S^n}$$

$$[S^m \multimap (S^m \otimes (S^n \multimap (S^n \otimes A)))] \otimes S^{m+n} \longrightarrow (S^{m+n} \otimes A)$$

$$\downarrow$$

$$[S^m \otimes (S^n \multimap (S^n \otimes A))] \otimes S^n$$

$$\downarrow$$

$$S^m \otimes (S^n \otimes A) \longrightarrow (S^{m+n} \otimes A)$$

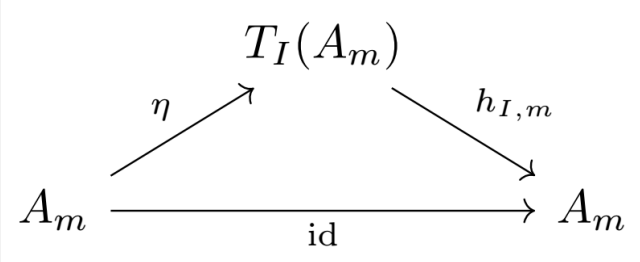
Fujii, Soichiro, Shin-ya Katsumata, and Paul-André Mellies. "Towards a formal theory of graded monads." *International Conference on Foundations of Software Science and Computation Structures*. Springer, Berlin, Heidelberg, 2016.

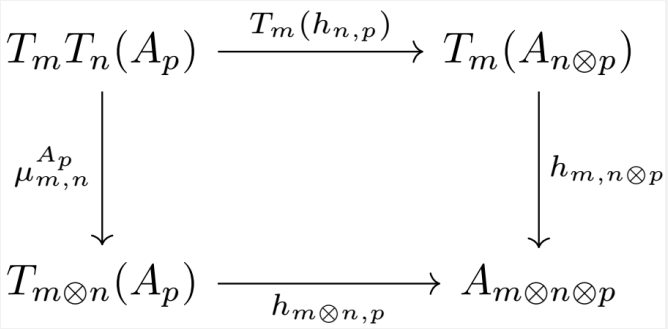
Graded algebras

A graded T-algebra is a pair (A, h) consisting of a functor $A : \mathcal{M} \to \mathbf{C}$ and a family h of morphisms

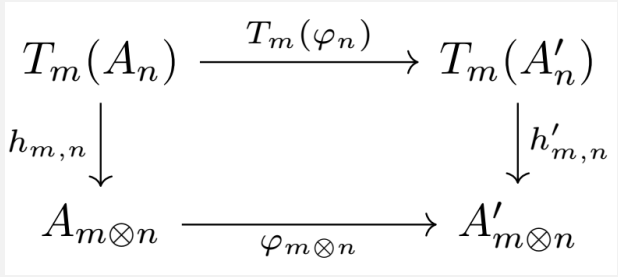
$$h_{m,n} : T_m(A_n) \longrightarrow A_{m \otimes n}$$

natural in m, n and such the following commute:





A homomorphism $\varphi : (A, h) \longrightarrow (A', h')$ of graded T-algebras is a natural transformation $\varphi : A \Rightarrow A'$ making the following commute for all m, n in $\mathcal{M}$:



The category $\mathbf{C}^T$ (Eilenberg-Moore construction) has graded T-algebras as objects and homomorphism between them as morphisms.

Formal graded monads

Street (1972): notion of monad for an arbitrary 2-category.

Repeat for graded monads to describe $\mathbf{C}^T$ as a universal construction.

Definition 10. *A* monad $\mathbf{T}$ *in* K *is given by a 0-cell* k*, a 1-cell* $T : k \to k$*, and 2-cells* $\mu : T \circ T \Rightarrow T$ *of* K *and* $\eta : \mathrm{id}_k \Rightarrow T$*, satisfying the usual axioms* $\mu \circ \eta T = \mathrm{id}_T = \mu \circ T\eta$ *and* $\mu \circ T\mu = \mu \circ \mu T$*.*

Now define a 2-category K, where the category $\mathbf{C}^T$ of graded algebras arises as an Eilenberg-Moore object in it. (Similarly for Kleisli).

Fujii, Soichiro, Shin-ya Katsumata, and Paul-André Mellies. "Towards a formal theory of graded monads." *International Conference on Foundations of Software Science and Computation Structures*. Springer, Berlin, Heidelberg, 2016.

Definition 11. *The* Eilenberg-Moore object *of* $\mathbf{T}$ *is a 0-cell* $k^{\mathbf{T}}$ *such that there is a family of isomorphisms of categories*

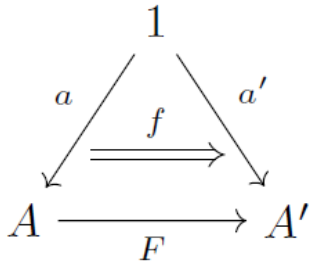
$$K(x, k^{\mathbf{T}}) \quad \cong \quad K(x, k)^{K(x, \mathbf{T})}$$

2-natural in $x \in K$. *Here the category on the right hand side is the (usual) Eilenberg-Moore category of the monad* $K(x, \mathbf{T})$ *on the category* $K(x, k)$.

We define the 2-category E^{++}:

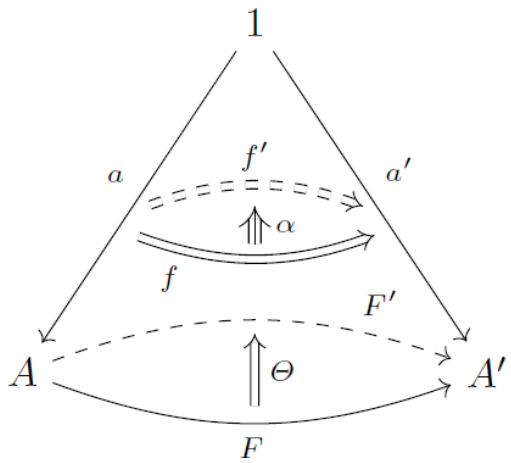
- 0-cells are 2-functors $a : 1 \to A$ where A is a 2-category. (Equivalently it is a pair (A,a) where A is a 0-cell A).

- 1-cells from (A, a) to (A', a') is a diagram in 2-Cat

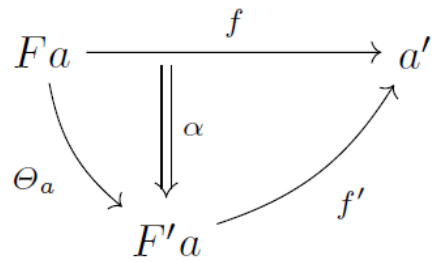


filled with a 2-natural transformation f. (Equivalently it is a pair (F, f) where $f : Fa \to a'$ is a 1-cell of A').

- a 2-cell from (F, f) to (F', f') is a diagram in 2-Cat:



filled with a modification α. (Equivalently, it is a pair (Θ, α) where α is a 2-cell of A'):



Theorem 16. *There is a family of isomorphisms of categories*

$$E^{++}((X,x),(\mathscr{M}\text{-}Cat,\mathscr{A}^{\mathbf{T}})) \quad \cong \quad E^{++}((X,x),(Cat,\mathscr{A}))^{E^{++}((X,x),\mathbf{T})}$$

Back to effects

Kleisli morphisms

Handler for exceptions

Conclusion

Monads	• Examples • Effects/Pure
Graded monads	• Degree of effectfulness
Algebras and characterisation	• EM and Kleisi constructions for graded monads