Relating Idioms, Arrows and Monads from Monoidal Adjunctions @ SYCO I

Exequiel Rivas

September 2018

Team πr^2, Inria

Semantics of effectful programming languages

The basic idea behind the semantics of programs described below is that a program denotes a morphism from A to TB.

E. Moggi 1989

Semantics of effectful programming languages

> *The basic idea behind the semantics of programs described below is that a program denotes a morphism from A to TB.*
>
> E. Moggi 1989

Moggi used *monads* for an unified treatment of effects.

$$\mathsf{Id} \xrightarrow{\eta} T \xleftarrow{\mu} T \circ T$$

His usages follows:

- η lifts values to effectful computations, i.e. `return`.
- μ composes two effects sequentially, i.e. `;`.

Wadler: monads as an interface

Monads can be *internalised* as an *interface*.

$$\textbf{class}\ \mathit{Functor}\ m \Rightarrow \mathit{Monad}\ m\ \textbf{where}$$
$$\quad \mathit{return} :: a \rightarrow m\ a$$
$$\quad (\gg\!=) :: m\ a \rightarrow (a \rightarrow m\ b) \rightarrow m\ b$$

The state monad *State* comes with operations

$$\mathit{get} :: \mathit{State}\ \mathit{Int} \quad , \quad \mathit{put} :: \mathit{Int} \rightarrow \mathit{State}\ ()$$

Computaions written using these operations and the interface.

$$\mathit{get} \gg\!= \lambda i \rightarrow \textbf{if}\ i \equiv 0\ \textbf{then}\ \mathit{return}\ \mathit{False}$$
$$\textbf{else}\ \mathit{put}\ 1 \gg\!= \backslash\_ \rightarrow \mathit{return}\ \mathit{True}$$

Arrows and applicative functors

Monads (as interfaces) has been generalised...

Arrows and applicative functors

Monads (as interfaces) has been generalised...

Providing more control over the computations.

$$
\begin{aligned}
&\textbf{class } \mathit{Functor}\ f \Rightarrow \mathit{Idiom}\ f \textbf{ where} \\
&\quad \mathit{pure} :: a \rightarrow f\ a \\
&\quad (\circledast)\ :: f\ (a \rightarrow b) \rightarrow f\ a \rightarrow f\ b
\end{aligned}
$$

Arrows and applicative functors

Monads (as interfaces) has been generalised...

Providing more control over the computations.

$$
\begin{array}{l}
\textbf{class } \mathit{Functor}\ f \Rightarrow \mathit{Idiom}\ f\ \textbf{where} \\
\quad \mathit{pure} :: a \to f\ a \\
\quad (\circledast)\ :: f\ (a \to b) \to f\ a \to f\ b
\end{array}
$$

$$
\begin{array}{l}
\textbf{class } \mathit{Arrow}\ (\leadsto)\ \textbf{where} \\
\quad \mathit{arr} \quad :: (x \to y) \to x \leadsto y \\
\quad (\ggg) :: (x \leadsto y) \to (y \leadsto z) \to x \leadsto z \\
\quad \mathit{first} \ :: (x \leadsto y) \to (x, z) \leadsto (y, z)
\end{array}
$$

Idioms are oblivious, arrows are meticulous, monads are ...

Lindley, Wadler and Yallop (2008), proved the equivalences

$$\text{Idiom} = \text{Arrow} + (x \rightsquigarrow y \cong 1 \rightsquigarrow (x \to y)),$$
$$\text{Monad} = \text{Arrow} + (x \rightsquigarrow y \cong x \to (1 \rightsquigarrow y))$$

Idioms are oblivious, arrows are meticulous, monads are . . .

Lindley, Wadler and Yallop (2008), proved the equivalences

$$\text{Idiom} = \text{Arrow} + (x \leadsto y \cong 1 \leadsto (x \to y)),$$
$$\text{Monad} = \text{Arrow} + (x \leadsto y \cong x \to (1 \leadsto y))$$

Following a syntactic approach: calculi and translations.

Idioms are oblivious, arrows are meticulous, monads are ...

Lindley, Wadler and Yallop (2008), proved the equivalences

$$\begin{aligned} \text{Idiom} &= \text{Arrow} + (x \rightsquigarrow y \cong 1 \rightsquigarrow (x \to y)), \\ \text{Monad} &= \text{Arrow} + (x \rightsquigarrow y \cong x \to (1 \rightsquigarrow y)) \end{aligned}$$

Following a syntactic approach: calculi and translations.

We aim for a semantic explanation, modelling:

- Arrows as profunctors $\mathbb{F}^{\mathsf{op}} \times \mathbb{F} \to \mathbb{S}$ with monoid structure.
- Monads and idioms as functors $\mathbb{F} \to \mathbb{S}$ with monoid structure.

Notions of computations as monoids

Monads, idioms and arrows have

- an operation embedding pure values: *return*, *pure* and *arr*.
- an operation sequencing computations: $(\gg\!=)$, $(\circledast)$ and $(\ggg)$.

Notions of computations as monoids

Monads, idioms and arrows have

- an operation embedding pure values: *return*, *pure* and *arr*.
- an operation sequencing computations: $(\gg\!=)$, $(\circledast)$ and $(\ggg)$.

Resemble monoids.

Notions of computations as monoids

Monads, idioms and arrows have

- an operation embedding pure values: *return*, *pure* and *arr*.
- an operation sequencing computations: $(\gg\!=)$, $(\circledast)$ and $(\ggg)$.

Resemble monoids.

We model computational effects using *monoidal categories*.

$$\textit{Monad} \Rrightarrow \text{Monoid in } ([\mathbb{F}, \mathbb{S}], \circ)$$

$$\textit{Idiom} \Rrightarrow \text{Monoid in } ([\mathbb{F}, \mathbb{S}], \star)$$

$$\textit{Arrow} \Rrightarrow \text{Monoid in } ([\mathbb{F}^{\mathrm{op}} \times \mathbb{F}, \mathbb{S}]_{\mathrm{s}}, \otimes)$$

Monoidal structures: $\circ$

The category of finitary endofunctors $[\mathbb{F}, \mathbb{S}]$ has a *substitution* monoidal structure.

$$(F \circ G)X = \int^{Y} FY \times (Y \to GX)$$

The inclusion $i : \mathbb{F} \to \mathbb{S}$ acts as unit.

Monoidal structures: $\circ$

The category of finitary endofunctors $[\mathbb{F}, \mathbb{S}]$ has a *substitution* monoidal structure.

$$(F \circ G)X = \int^{Y} FY \times (Y \to GX)$$

The inclusion $i : \mathbb{F} \to \mathbb{S}$ acts as unit.

A monoid

$$i \xrightarrow{\textit{return}} M \xleftarrow{(\gg\!=)} M \circ M$$

in $([\mathbb{F}, \mathbb{S}], \circ, i)$ is a monad.

Monoidal structures: $\star$

The category $[\mathbb{F}, \mathbb{S}]$ also has a *convolution* monoidal structure.

$$(F \star G)X = \int^{Y} FY \times G(Y \to X)$$

The inclusion $i : \mathbb{F} \to \mathbb{S}$ also acts as the unit.

Monoidal structures: $\star$

The category $[\mathbb{F}, \mathbb{S}]$ also has a *convolution* monoidal structure.

$$(F \star G)X = \int^{Y} FY \times G(Y \to X)$$

The inclusion $i : \mathbb{F} \to \mathbb{S}$ also acts as the unit.

A monoid

$$i \xrightarrow{pure} F \xleftarrow{(\circledast)} F \star F$$

in $([\mathbb{F}, \mathbb{S}], \star, i)$ is an idiom.

Intermezzo: strong profunctors

Profunctors compatible with the underlying cartesian structure.

Intermezzo: strong profunctors

Profunctors compatible with the underlying cartesian structure.

Definition: strong profunctor

A profunctor $P : \mathbb{F}^{\mathsf{op}} \times \mathbb{F} \to \mathbb{S}$ is *strong* if it comes equipped with a family of morphisms

$$\mathrm{str}_{X,Y,Z} : P(X, Y) \to P(X \times Z, Y \times Z)$$

natural in X, Y and dinatural in Z such that the following equations hold:

$$P(\mathrm{id}, \pi_1) \circ \mathrm{str}_{X,Y,1} = P(\pi_1, \mathrm{id}),$$
$$\mathrm{str}_{X,Y,W} \circ \mathrm{str}_{X,Y,V} = P(\alpha^{-1}, \alpha) \circ \mathrm{str}_{X,Y,V \times W}$$

Monoidal structures: $\otimes$

Strong profunctors $\mathbb{F}^{\mathsf{op}} \times \mathbb{F} \to \mathbb{S}$ have composition of profunctors.

$$(P \otimes Q)(X, Y) = \int^{W} P(X, W) \times Q(W, Y)$$

The hom-set $\mathrm{Hom}_{\mathbb{F}} : \mathbb{F}^{\mathsf{op}} \times \mathbb{F} \to \mathbb{S}$ as the unit.

Monoidal structures: $\otimes$

Strong profunctors $\mathbb{F}^{op} \times \mathbb{F} \to \mathbb{S}$ have composition of profunctors.

$$(P \otimes Q)(X, Y) = \int^{W} P(X, W) \times Q(W, Y)$$

The hom-set $\mathrm{Hom}_{\mathbb{F}} : \mathbb{F}^{op} \times \mathbb{F} \to \mathbb{S}$ as the unit.

A monoid

$$\mathrm{Hom}_{\mathbb{F}} \xrightarrow{\;arr\;} A \xleftarrow{\;(\ggg)\;} A \otimes A$$

in $([\mathbb{F}^{op} \times \mathbb{F}, \mathbb{S}]_s, \otimes, \mathrm{Hom}_{\mathbb{F}})$ is an arrow.

The equations II

$$\text{Idiom} = \text{Arrow} + (x \rightsquigarrow y \cong 1 \rightsquigarrow (x \to y)),$$
$$\text{Monad} = \text{Arrow} + (x \rightsquigarrow y \cong x \to (1 \rightsquigarrow y))$$

We have defined *Idiom*, *Monad* and *Arrow* in our model:

$$\mathit{Monad} \Rrightarrow \text{Monoid in } ([\mathbb{F}, \mathbb{S}], \circ)$$
$$\mathit{Idiom} \Rrightarrow \text{Monoid in } ([\mathbb{F}, \mathbb{S}], \star)$$
$$\mathit{Arrow} \Rrightarrow \text{Monoid in } ([\mathbb{F}^{\mathrm{op}} \times \mathbb{F}, \mathbb{S}]_{\mathrm{s}}, \otimes)$$

The equations II

$$\text{Idiom} = \text{Arrow} + (x \rightsquigarrow y \cong 1 \rightsquigarrow (x \to y)),$$
$$\text{Monad} = \text{Arrow} + (x \rightsquigarrow y \cong x \to (1 \rightsquigarrow y))$$

We have defined *Idiom*, *Monad* and *Arrow* in our model:

$$\textit{Monad} \Rrightarrow \text{Monoid in } ([\mathbb{F}, \mathbb{S}], \circ)$$
$$\textit{Idiom} \Rrightarrow \text{Monoid in } ([\mathbb{F}, \mathbb{S}], \star)$$
$$\textit{Arrow} \Rrightarrow \text{Monoid in } ([\mathbb{F}^{\mathrm{op}} \times \mathbb{F}, \mathbb{S}]_{\mathrm{s}}, \otimes)$$

Isomorphisms on the right still missing.

Formalising the isomorphisms

As a first step, we model the isomorphisms for profunctors.

If A is the strong profunctor underlying the arrow $(\rightsquigarrow)$

$$x \rightsquigarrow y \cong 1 \rightsquigarrow (x \to y) \quad \Rrightarrow \quad A(x, y) \cong A(1, x \to y),$$
$$x \rightsquigarrow y \cong x \to (1 \rightsquigarrow y) \quad \Rrightarrow \quad A(x, y) \cong ix \to A(1, y).$$

Formalising the isomorphisms

As a first step, we model the isomorphisms for profunctors.

If A is the strong profunctor underlying the arrow $(\rightsquigarrow)$

$$x \rightsquigarrow y \cong 1 \rightsquigarrow (x \to y) \quad \Rrightarrow \quad A(x, y) \cong A(1, x \to y),$$
$$x \rightsquigarrow y \cong x \to (1 \rightsquigarrow y) \quad \Rrightarrow \quad A(x, y) \cong ix \to A(1, y).$$

We try to factorise

$$A(1, x \to y) \qquad \text{and} \qquad ix \to A(1, y)$$

as functors applied to A on x and y.

Fixing one parameter

A strong profunctor in $[\mathbb{F}^{\mathrm{op}} \times \mathbb{F}, \mathbb{S}]_{\mathrm{s}}$ can be mapped to a functor $\mathbb{F} \to \mathbb{S}$ by evaluating its first parameter.

Fixing one parameter

A strong profunctor in $[\mathbb{F}^{\mathrm{op}} \times \mathbb{F}, \mathbb{S}]_{\mathrm{s}}$ can be mapped to a functor $\mathbb{F} \to \mathbb{S}$ by evaluating its first parameter.

In particular, evaluating with 1, we obtain

$$\begin{array}{rcl} \text{-}^{*} & : & [\mathbb{F}^{\mathrm{op}} \times \mathbb{F}, \mathbb{S}]_{\mathrm{s}} \longrightarrow [\mathbb{F}, \mathbb{S}] \\ A^{*} & = & Z \mapsto A(1, Z) \\ \tau^{*}{}_{Z} & = & \tau_{1,Z} \end{array}$$

From functors to strong profunctors

The functor $-^*$ has left and right adjoints:

From functors to strong profunctors

The functor $-^*$ has left and right adjoints:

$$
\begin{aligned}
-_! \quad &: \quad [\mathbb{F}, \mathbb{S}] \longrightarrow [\mathbb{F}^{\mathrm{op}} \times \mathbb{F}, \mathbb{S}]_{\mathrm{s}} \\
F_! \quad &= \quad (X, Y) \mapsto F(X \to Y)
\end{aligned}
$$

From functors to strong profunctors

The functor $-^*$ has left and right adjoints:

$$\begin{aligned} -_! &: [\mathbb{F}, \mathbb{S}] \longrightarrow [\mathbb{F}^{\mathrm{op}} \times \mathbb{F}, \mathbb{S}]_{\mathrm{s}} \\ F_! &= (X, Y) \mapsto F(X \to Y) \end{aligned}$$

$$\begin{aligned} -_* &: [\mathbb{F}, \mathbb{S}] \longrightarrow [\mathbb{F}^{\mathrm{op}} \times \mathbb{F}, \mathbb{S}]_{\mathrm{s}} \\ F_* &= (X, Y) \mapsto i\ X \to F\ Y \end{aligned}$$

From functors to strong profunctors

The functor $-^*$ has left and right adjoints:

$$\begin{aligned} -_! \quad &: \quad [\mathbb{F}, \mathbb{S}] \longrightarrow [\mathbb{F}^{\mathrm{op}} \times \mathbb{F}, \mathbb{S}]_{\mathrm{s}} \\ F_! \quad &= \quad (X, Y) \mapsto F(X \to Y) \end{aligned}$$

$$\begin{aligned} -_* \quad &: \quad [\mathbb{F}, \mathbb{S}] \longrightarrow [\mathbb{F}^{\mathrm{op}} \times \mathbb{F}, \mathbb{S}]_{\mathrm{s}} \\ F_* \quad &= \quad (X, Y) \mapsto i\ X \to F\ Y \end{aligned}$$

We end up with an adjoint triple

$$-_! \quad \dashv \quad -^* \quad \dashv \quad -_*$$

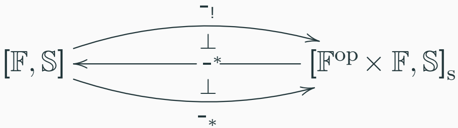
The picture

We obtain the diagram

$$[\mathbb{F},\mathbb{S}] \;\; \begin{matrix} \xrightarrow{\quad -_! \quad} \\ \perp \\ \xleftarrow{\quad -^* \quad} \\ \perp \\ \xrightarrow[\quad -_* \quad]{} \end{matrix} \;\; [\mathbb{F}^{\mathrm{op}} \times \mathbb{F}, \mathbb{S}]_{\mathrm{s}}$$

The picture

We obtain the diagram



and the isomorphisms become

$$A(x, y) \cong A(1, x \to y) \quad \Rrightarrow \quad A \cong (A^*)_!$$
$$A(x, y) \cong ix \to A(1, y) \quad \Rrightarrow \quad A \cong (A^*)_*$$

What about the monoidal structures?

$$\text{Idiom} = \text{Arrow} + (x \rightsquigarrow y \cong 1 \rightsquigarrow (x \to y)),$$
$$\text{Monad} = \text{Arrow} + (x \rightsquigarrow y \cong x \to (1 \rightsquigarrow y)).$$

What about the monoidal structures?

$$\mathsf{Idiom} = \mathsf{Arrow} + (x \rightsquigarrow y \cong 1 \rightsquigarrow (x \to y)),$$
$$\mathsf{Monad} = \mathsf{Arrow} + (x \rightsquigarrow y \cong x \to (1 \rightsquigarrow y)).$$

On the isomorphisms we only dealt with the objects.

What about the monoidal structures?

$$\text{Idiom} = \text{Arrow} + (x \rightsquigarrow y \cong 1 \rightsquigarrow (x \to y)),$$
$$\text{Monad} = \text{Arrow} + (x \rightsquigarrow y \cong x \to (1 \rightsquigarrow y)).$$

On the isomorphisms we only dealt with the objects.

Theorem

For an adjoint triple $F \dashv G \dashv H$, we have that the comonad FG and the monad HG are adjoint $FG \dashv HG$.

What about the monoidal structures?

$$\text{Idiom} = \text{Arrow} + (x \rightsquigarrow y \cong 1 \rightsquigarrow (x \to y)),$$
$$\text{Monad} = \text{Arrow} + (x \rightsquigarrow y \cong x \to (1 \rightsquigarrow y)).$$

On the isomorphisms we only dealt with the objects.

Theorem

For an adjoint triple $F \dashv G \dashv H$, we have that the comonad FG and the monad HG are adjoint $FG \dashv HG$.

From the adjoint triple

$$-_! \quad \dashv \quad -^* \quad \dashv \quad -_*$$

we obtain

$$(-^*)_! = \Box \dashv \Diamond = (-^*)_*$$

Idempotent monads and monoids

In our case, the comonad $\Box$ and the monad $\Diamond$ are idempotent.

Idempotent monads and monoids

In our case, the comonad $\Box$ and the monad $\Diamond$ are idempotent.

Definition: T-monoid

If $T : C \to C$ is an idempotent (co)monad, then a T-monoid is quadruple (M, m, e, α) where

- $(M, m : M \otimes M \to M, e : I \to M)$ is a monoid;
- $(M, \alpha : TM \to M)$ is a T-algebra.

T-monoids form a category $\mathrm{Mon}(T)$.

Idempotent monads and monoids

In our case, the comonad $\Box$ and the monad $\Diamond$ are idempotent.

Definition: T-monoid

If $T : C \to C$ is an idempotent (co)monad, then a T-monoid is quadruple (M, m, e, α) where

- $(M, m : M \otimes M \to M, e : I \to M)$ is a monoid;
- $(M, \alpha : TM \to M)$ is a T-algebra.

T-monoids form a category $\mathrm{Mon}\,(T)$.

For idempotent (co)monads, (co)algebras are isos. A T-monoid (M, m, e, α) is a

$$\text{Monoid on } C + (M \cong TM)$$

The equivalences

$$\mathsf{Idiom} = \mathsf{Arrow} + (x \rightsquigarrow y \cong 1 \rightsquigarrow (x \to y))$$

$$\Downarrow$$

$\mathrm{Mon}\left([\mathbb{F}, \mathbb{S}]\right)$ and $\mathrm{Mon}\left(\square\right)$ are equivalent categories.

The equivalences

$$\text{Idiom} = \text{Arrow} + (x \leadsto y \cong 1 \leadsto (x \to y))$$

$$\Downarrow$$

$\mathrm{Mon}\,([\mathbb{F}, \mathbb{S}])$ and $\mathrm{Mon}\,(\Box)$ are equivalent categories.

$$\text{Monad} = \text{Arrow} + (x \leadsto y \cong x \to (1 \leadsto y))$$

$$\Downarrow$$

$\mathrm{Mon}\,([\mathbb{F}, \mathbb{S}])$ and $\mathrm{Mon}\,(\Diamond)$ are equivalent categories.

Proof sketch I

To prove

$$\mathrm{Mon}\,([\mathbb{F},\mathbb{S}]) \text{ and } \mathrm{Mon}\,(\square) \text{ are equivalent categories}$$

Proof sketch I

To prove

$$\mathrm{Mon}\left([\mathbb{F},\mathbb{S}]\right) \text{ and } \mathrm{Mon}\left(\Box\right) \text{ are equivalent categories}$$

note that both functors are monoidal (monoidal adjunction)

$$\left([\mathbb{F},\mathbb{S}],\star\right) \underset{-^{*}}{\overset{-_{!}}{\underset{\bot}{\rightleftarrows}}} \left([\mathbb{F}^{\mathrm{op}}\times\mathbb{F},\mathbb{S}]_{\mathrm{s}},\otimes\right)$$

Proof sketch I

To prove

$$\mathrm{Mon}\left([\mathbb{F},\mathbb{S}]\right) \text{ and } \mathrm{Mon}\left(\square\right) \text{ are equivalent categories}$$

note that both functors are monoidal (monoidal adjunction)

$$\left([\mathbb{F},\mathbb{S}]\,,\star\right) \underset{-^{*}}{\overset{-_{!}}{\underset{\bot}{\rightleftarrows}}} \left([\mathbb{F}^{\mathrm{op}}\times\mathbb{F},\mathbb{S}]_{\mathrm{s}},\otimes\right)$$

Functors lift to categories of monoids.

Proof sketch II

In the case

$$\mathrm{Mon}\left([\mathbb{F}, \mathbb{S}]\right) \text{ and } \mathrm{Mon}\left(\Diamond\right) \text{ are equivalent categories}$$

Proof sketch II

In the case

$$\mathrm{Mon}\left([\mathbb{F}, \mathbb{S}]\right) \text{ and } \mathrm{Mon}\left(\Diamond\right) \text{ are equivalent categories}$$

the adjunction

$$\left([\mathbb{F}^{\mathrm{op}} \times \mathbb{F}, \mathbb{S}]_{\mathrm{s}}, \otimes\right) \underset{-_*}{\overset{-^*}{\rightleftarrows}} \left([\mathbb{F}, \mathbb{S}], \circ\right) \quad \bot$$

is a monoidal conjunction. No guarantees that $-^*$ preserves monoids.

Proof sketch II

In the case

$$\mathrm{Mon}\left([\mathbb{F}, \mathbb{S}]\right) \text{ and } \mathrm{Mon}\left(\Diamond\right) \text{ are equivalent categories}$$

the adjunction

$$\left([\mathbb{F}^{\mathrm{op}} \times \mathbb{F}, \mathbb{S}]_{\mathrm{s}}, \otimes\right) \underset{\text{-}_*}{\overset{\text{-}^*}{\rightleftarrows}} \left([\mathbb{F}, \mathbb{S}], \circ\right) \quad \bot$$

is a monoidal conjunction. No guarantees that -* preserves monoids.

A result by Porst and Street gives conditions when an opmonoidal functor preserves monoids.

Conclusions

We have extended the notions of computation as monoids view to show a semantic counterpart to Lindley et al.'s result.

Conclusions

We have extended the notions of computation as monoids view to show a semantic counterpart to Lindley et al.'s result.

Further work includes

- replacing $\mathbb{F}$ and $\mathbb{S}$.
- relating to relative monads and other solutions that do not suffer of size issues.
- seeing how comonads and other notions fit in the picture.