Monads and More: Part 2

Tarmo Uustalu, Tallinn

Nottingham, 14–18 May 2007

Monads from adjuctions (Huber)

- For any pair of adjoint functors $L : \mathcal{C} \to \mathcal{D}$, $R : \mathcal{D} \to \mathcal{C}$, $L \dashv R$ with unit $\eta : \mathsf{Id}_{\mathcal{C}} \stackrel{\cdot}{\to} RL$ and counit $\varepsilon : LR \stackrel{\cdot}{\to} \mathsf{Id}_{\mathcal{D}}$, the functor RL carries a monad structure defined by
  - $\eta^{RL} =_{\mathrm{df}} \mathsf{Id} \xrightarrow{\eta} RL$,
  - $\mu^{RL} =_{\mathrm{df}} RLRL \xrightarrow{R\varepsilon L} RL$.
- The Kleisli and Eilenberg-Moore adjunctions witness that any monad on $\mathcal{C}$ admits a factorization like this.

Examples

- Side-effect monad:
  - $L, R : \mathcal{C} \to \mathcal{C}$, $LA =_{\mathrm{df}} A \times S$, $RB =_{\mathrm{df}} S \Rightarrow B$,

$$\frac{A \times S \to B}{A \to S \Rightarrow B}$$

  - $RLA = S \Rightarrow A \times S$,
- An exotic one:
  - $L, R : \mathcal{C} \to \mathcal{C}$, $LA =_{\mathrm{df}} \mu X.A + X \times S \cong A \times \mathrm{List} S$, $RB =_{\mathrm{df}} \nu Y.B \times (S \Rightarrow Y)$,

$$\frac{\mu X.A + X \times S \to B}{A \to \nu Y.B \times (S \Rightarrow Y)}$$

  - $RLA = \nu Y.(\mu X.A + X \times S) \times (S \Rightarrow Y) \cong \nu Y.A \times \mathrm{List} S \times (S \Rightarrow Y)$.
  - What notion of computation does this correspond to?

- Continuations monad:
  - $L : \mathcal{C} \rightarrow \mathcal{C}^{\mathrm{op}}$, $LA =_{\mathrm{df}} A \Rightarrow E$,
    $R : \mathcal{C}^{\mathrm{op}} \rightarrow \mathcal{C}$, $RB =_{\mathrm{df}} B \Rightarrow E$,

$$\dfrac{\dfrac{\dfrac{A \Rightarrow E \leftarrow B}{E \leftarrow B \times A}}{A \times B \rightarrow E}}{A \rightarrow B \Rightarrow E}$$

  - $RLA = (A \Rightarrow E) \Rightarrow E$.

Monads from adjunctions ctd.

- Given two functors $L : \mathcal{C} \to \mathcal{D}$ and $R : \mathcal{D} \to C$, $L \dashv R$ and a monad T on $\mathcal{D}$, we obtain that RTL is a monad on $\mathcal{C}$.
- This is because T factorizes as UJ where $J \vdash U$ is the Kleisli adjunction.
  That means an adjoint situation $JL \vdash RU$ implying that $RUJL = RTL$ is a monad.
- The monad structure is
  - $\eta^{RTL} =_{\mathrm{df}} \mathsf{Id} \xrightarrow{\eta} RL \xrightarrow{R\eta^T L} RTL$,
  - $\mu^{RTL} =_{\mathrm{df}} RTLRTL \xrightarrow{RT\varepsilon TL} RTTL \xrightarrow{\mu^T} RTL$.

Examples

- State monad transformer:
  - $L, R : \mathcal{C} \to \mathcal{C}$, $LA =_{\mathrm{df}} A \times S$, $RB =_{\mathrm{df}} S \Rightarrow B$,
  - T – a monad on $\mathcal{C}$,
  - $RTLA = S \Rightarrow T(A \times S)$,
  - In particular, for T the exceptions monad we get $RTLA = S \Rightarrow (A \times S) + E$.
- Continuations monad transformer:
  - $L : \mathcal{C} \to \mathcal{C}^{\mathrm{op}}$, $LA =_{\mathrm{df}} A \Rightarrow E$,
    $R : \mathcal{C}^{\mathrm{op}} \to \mathcal{C}$, $RB =_{\mathrm{df}} B \Rightarrow E$,
  - T – a monad on $\mathcal{C}^{\mathrm{op}}$, i.e., a comonad on $\mathcal{C}$,
  - $RTLA =_{\mathrm{df}} T(A \Rightarrow E) \to E$.

Free algebras

- Given a endofunctor H on a category $\mathcal{C}$, the initial algebra of $(H^*A, [\eta_A, \tau_A])$ of $A + H-$ (if it exists) is the type of wellfounded H-trees with mutable leaves from A, i.e., H-terms over variables from A.
- $((H^*A, \tau_A), \eta_A))$ is the free H-algebra on A.
- (H^*, η, μ) is a monad where μ flattens a tree whose mutable leaves are trees into a tree, i.e., a term over terms into a term.
- $((H^*, \eta, \mu), \tau)$ is the free monad on H.
- The final coalgebras $H^\infty A$ of $A + H-$ for each A also a give a monad.

Monads from parameterized monads via initial algebras / final coalgebras (U.)

- A *parameterized monad* on $\mathcal{C}$ is a functor $F : \mathcal{C} \to \mathbf{Monad}(\mathcal{C})$.
- If F is a parameterized monad then the functors $T, T^\infty : \mathcal{C} \to \mathcal{C}$ defined by $TA =_{\mathrm{df}} \mu X.FXA$ and $T^\infty A =_{\mathrm{df}} \nu X.FXA$ carry a monad structure.
- In fact more can be said about them, but here we won't.

Examples

- Free monads:
  - $FXA =_{\mathrm{df}} A + HX$ where $H : \mathcal{C} \to \mathcal{C}$,
  - $TA =_{\mathrm{df}} \mu X.A + HX$, $T^\infty A =_{\mathrm{df}} \nu X.A + HX$.
  - These are the types of wellfounded/nonwellfounded H-trees with mutable leaves from A.
- Rose tree types:
  - $FXA =_{\mathrm{df}} A \times HX$ where $H : \mathcal{C} \to \mathbf{Monoid}(\mathcal{C})$,
  - $TA =_{\mathrm{df}} \mu X.A \times HX$, $T^\infty A =_{\mathrm{df}} \nu X.A + HX$.
  - If $HX =_{\mathrm{df}} \mathrm{List}X$, these are the types of wellfounded/nonwellfounded A-labelled rose trees.

- Types of hyperfunctions with a fixed domain:
  - $FXA =_{\mathrm{df}} HX \Rightarrow A$ where $H : \mathcal{C} \rightarrow \mathcal{C}^{\mathrm{op}}$,
  - $TA =_{\mathrm{df}} \mu X.HX \Rightarrow A$, $T^{\infty}A =_{\mathrm{df}} \nu X.HX \Rightarrow A$.
  - If $HX =_{\mathrm{df}} X \Rightarrow E$, these are the types of wellfounded/nonwellfounded hyperfunctions from E to A. (Of course these types do no exist in **Set**.)

Distributive laws

- If T, S are monads on $\mathcal{C}$, it is not generally the case that ST is a monad. But sometimes it is.
- A *distributive law* of a monad T over a monad S is a natural transformation $\lambda : TS \stackrel{\cdot}{\to} ST$ satisfying

$$\begin{array}{ccc} T & = & T \\ {\scriptstyle T\eta^S}\downarrow & & \downarrow{\scriptstyle \eta^S T} \\ TS & \xrightarrow[\lambda]{} & ST \end{array} \qquad \begin{array}{ccccc} TSS & \xrightarrow{\lambda S} & STS & \xrightarrow{S\lambda} & SST \\ {\scriptstyle T\mu^S}\downarrow & & & & \downarrow{\scriptstyle \mu^S T} \\ TS & & \xrightarrow[\lambda]{} & & ST \end{array}$$

$$\begin{array}{ccc} S & = & S \\ {\scriptstyle \eta^T S}\downarrow & & \downarrow{\scriptstyle S\eta^T} \\ TS & \xrightarrow[\lambda]{} & ST \end{array} \qquad \begin{array}{ccccc} TTS & \xrightarrow{T\lambda} & TST & \xrightarrow{\lambda T} & STT \\ {\scriptstyle \mu^T S}\downarrow & & & & \downarrow{\scriptstyle S\mu^T} \\ TS & & \xrightarrow[\lambda]{} & & ST \end{array}$$

- A distributive law λ of T over S gives a monad structure on the endofunctor ST:
  - $\eta^{ST} =_{\mathrm{df}} \mathsf{Id} \xrightarrow{\eta^S \eta^T} ST$,
  - $\mu^{ST} =_{\mathrm{df}} STST \xrightarrow{S\lambda T} SSTT \xrightarrow{\mu^S \mu^T} ST$.

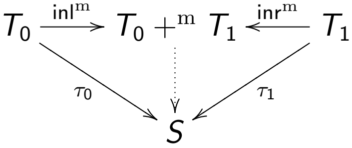
Examples

- The exceptions monad distributes over any monad.
  - S – a monad,
  - $TA =_{\mathrm{df}} A + E$ where E is an object,
  - $\lambda =_{\mathrm{df}} SA + E \xrightarrow{\mathsf{id}+\eta^S} SA + SE \xrightarrow{[S\mathsf{inl}, S\mathsf{inr}]} S(A+E)$,
  - $STA = S(A+E)$.
  - For T the state monad, this gives $ST = S \Rightarrow (A+E) \times S$, which is a different combination of exceptions and state than we saw before.
- The output monad distributes over any $(1, \times)$ strong monad.
  - (S, sl) – a strong monad,
  - $TA =_{\mathrm{df}} A \times E$ where E is a monoid,
  - $\lambda =_{\mathrm{df}} SA \times E \xrightarrow{\mathsf{sr}} S(A \times E)$,
  - $STA = S(A \times E)$.

- Any $(1, \times)$ strong monad distributes over the environment monad.
    - (T, sl) – a strong monad,
    - $SA =_{\mathrm{df}} E \Rightarrow A$ where E is an object,
    - $\lambda =_{\mathrm{df}} \Lambda(T(E \Rightarrow A) \times A \xrightarrow{\mathsf{sr}} T((E \Rightarrow A) \times A) \xrightarrow{T\mathsf{ev}} E)$,
    - $STA = E \Rightarrow TA$.

Coproduct of monads

- An interesting way to combine monads is the coproduct of monads.
- A coproduct of two monads T_0 and T_1 on $\mathcal{C}$ is their coproduct in **Monad**$(\mathcal{C})$.
- I.e., it is a monad $T_0 +^{\mathrm{m}} T_1$ together with two monad maps $\mathsf{inl}^{\mathrm{m}} : T_0 \to^{\mathrm{m}} T_0 +^{\mathrm{m}} T_1$, $\mathsf{inr}^{\mathrm{m}} : T_0 \to^{\mathrm{m}} T_0 +^{\mathrm{m}} T_1$ such that for any monad S and monad maps $\tau_0 : T_0 \to^{\mathrm{m}} S$, $\tau_1 : T_1 \to^{\mathrm{m}} S$ there exists a unique map $T_0 +^{\mathrm{m}} T_1 \to^{\mathrm{m}} S$ satisfying



Coproduct of free monads

- The coproduct of the free monads of functors F, G is the free monad of their coproduct:

$$F^\star +^{\mathrm{m}} G^\star = (F + G)^*$$

(obvious, since the free monad delivering functor has a left adjoint and hence preserves colimits).

- More generally, the coproduct of a free monad F^* with an arbitary monad S is this (if $(FS)^*$ exists):

$$F^* +^{\mathrm{m}} S = S(FS)^*$$

i.e.,

$$(F^* +^{\mathrm{m}} S)A = S(\mu X.A + FSX) = \mu X.S(A + FX)$$

Ideal monads (Adámek)

- An *ideal monad* on $\mathcal{C}$ is a monad (T, η, μ) together with an endofunctor T' on $\mathcal{C}$ and a natural transformation $\mu' : T'T \stackrel{\cdot}{\to} T'$ such that
  - $T = \mathsf{Id} + T'$,
  - $\eta = \mathsf{inl}$,
  - $\mu = [id, \mathsf{inr} \circ \mu']$.

Coproduct of ideal monads (Ghani, U.)

- Given two ideal monads $R = \mathsf{Id} + R'$ and $S = \mathsf{Id} + S'$, their coproduct is an ideal monad $T = \mathsf{Id} + T_0 + T_1$ where

$$(T_0 A, T_1 A) =_{\mathrm{df}} \mu(X, Y).(R'(A + Y)), S'(A + X))$$