On Retracts of Algebras with Iteration

Sergey Goncharov\textsuperscript{a}, Lutz Schröder\textsuperscript{a}, Christoph Rauch\textsuperscript{a}, Maciej Piróg\textsuperscript{b}

CMCS 2018, 14-15 April 2018, Thessaloniki, Greece

\textsuperscript{a}Friedrich-Alexander-Universität Erlangen-Nürnberg
\textsuperscript{b}Uniwersytet Wrocławski
• (complete) Elgot monads $\subseteq$ guarded Elgot monads $\subseteq$ guarded (co-Cartesian) iteration categories $\subseteq$ guarded traced categories$^1$

$^1$This FoSSaCS: Goncharov and Schröder 2018, Guarded Traced Categories
A Hierarchy of Structures for Iteration

- (complete) Elgot monads $\subseteq$ guarded Elgot monads $\subseteq$ guarded (co-Cartesian) iteration categories $\subseteq$ guarded traced categories$^1$

- What about algebras?

---

$^1$This FoSSaCS: Goncharov and Schröder 2018, Guarded Traced Categories
Elgot Monads for Computations
Monads: Semantic Perspective

- Monads formalize generalized functions $f : X \rightarrow TY$
  - e.g. nondeterministic ($T = \mathcal{P}X$)
  - e.g. partial (with $TX = X + 1$)
Monads: Semantic Perspective

- Monads formalize generalized functions $f : X \to TY$
  - e.g. nondeterministic ($T = \mathcal{P}X$)
  - e.g. partial (with $TX = X + 1$)

- $T$ is a type constructor, together with
  - unit $\eta : X \to TX$
  - Kleisli lifting $(f : X \to TY) \mapsto (f^* : TX \to TY)$

  inducing a category under

  \[ \text{id} = \eta : X \to TX \quad f \circ g = (f : Y \to TZ)^* \circ (g : X \to TY) \]

  In Haskell's point-full notation: do $x \leftarrow p; f(x) = f^*(p)$
• An iteration operator assigns $f^\dagger : X \rightarrow TY$ to $f : X \rightarrow T(Y + X)$
• An iteration operator assigns $f^+ : X \to TY$ to $f : X \to T(Y + X)$

• This yields a semantics for while loops:

  $$\text{while}(b : X \to \text{Bool}, p : X \to TX)$$

  $$= (\lambda x. \text{if } b(x) \text{ then } (T \text{ inr})(p(x)) \text{ else } (\text{inl})(x))^+$$

(e.g. $T$ is the partial store monad $(- \times S + 1)^S$)
An iteration operator assigns $f^\dagger : X \rightarrow TY$ to $f : X \rightarrow T(Y + X)$.

This yields a semantics for while loops:

$$\text{while}(b : X \rightarrow \text{Bool}, p : X \rightarrow TX) = (\lambda x. \text{if } b(x) \text{ then } (T \text{inr})(p(x)) \text{ else } (\eta \text{inl})(x))^\dagger \quad \text{(e.g. } T \text{ is the partial store monad } (\sim \times S + 1)^S)$$

Now, the theory of while-programs, featuring laws like

$$\text{while}(b, p)(x) = \text{if } b(x) \text{ then do } x \leftarrow p(x); \text{ while}(b, p)(x) \text{ else } \eta(x)$$

can be couched in terms of $(\sim)^\dagger$.
A monad $T$ is a (complete) Elgot monad\textsuperscript{2} if it is equipped with an operator $(f : X \to T(Y + X)) \leftrightarrow (f^\dagger : X \to TY)$ satisfying

**Fixpoint:**

$$f \circ f = f$$

**Naturality:**

$$f \circ g = g$$

\textsuperscript{2}Adámek, Milius, and Velebil, 2010, Equational properties of iterative monads
Elgot Monads (2/2)

**Codiodiagonal:**

\[
g : Y \to X \\
X \to g : Y \to X \\
g \circ \lambda : X \to Y \to X \\
= \\
g : Y \to X \\
X \to g : Y \to X \\
g \circ \lambda : X \to Y \to X \\
= \\
g : Y \to X \\
X \to g : Y \to X \\
\]

**Uniformity:**

\[
h : Z \to X \\
X \to f : Y \to X \\
f \circ \rho : Z \to X \\
= \\
h : Z \to X \\
X \to f : Y \to X \\
f \circ \rho : Z \to X \\
\]

\[
h : Z \to X \\
X \to f : Y \to X \\
f \circ \rho : Z \to X \\
\downarrow \\
h : Z \to X \\
X \to f : Y \to X \\
f \circ \rho : Z \to X \\
= \\
h : Z \to X \\
X \to f : Y \to X \\
f \circ \rho : Z \to X \\
\]

\[
h : Z \to X \\
X \to f : Y \to X \\
f \circ \rho : Z \to X \\
\]
Our recent work\(^3\) implies that Elgot monads are precisely characterized as iteration-congruent retracts:

\[ \rho : \nu \gamma \to \leftarrow T\gamma \leftrightarrow T : \nu \]

(\(\rho\) is a split epi monad morphism and \(\rho f = \rho g \implies \rho f^\dagger = \rho g^\dagger\)) where the left monad is **completely iterative**, i.e. iteration is partial, but unique.

---

\(^3\)Goncharov, Schröder, Rauch, and Piróg 2017, Unifying Guarded and Unguarded Iteration
Capretta’s delay monad:

- In Set

\[ \nu \gamma. X + \gamma \leftrightarrow X + 1 \]

\(^4\) Veltri 2017, A Type-Theoretical Study of Nontermination
Capretta’s delay monad:

- In Set

\[ \nu \gamma . X + \gamma \iff X + 1 \]

- In general, building

\[ DX = \nu \gamma . X + \gamma \iff \tilde{DX} \]

equalizing id : \( DX \to DX \) and later : \( DX \to DX \) requires choice principles like countable choice\(^4\)

---

\(^4\) Veltri 2017, A Type-Theoretical Study of Nontermination
Elgot Algebras for Data
Motivating Elgot Algebras

- Powerset monad $\mathcal{P}$ is Elgot, and
  - The free algebra $(\mathcal{P}1 \cong 2, \Diamond : \mathcal{P}2 \to 2)$ supports iteration $(f : X \to 2 + \mathcal{P}X) \mapsto (f^\dagger : X \to 2)$ inherited from $\mathcal{P}$
  - But also the dual $(\mathcal{P}1 \cong 2, \Box : \mathcal{P}2 \to 2)$ supports iteration, albeit not inherited from $\mathcal{P}$

This is relevant for weakest precondition semantics\(^5\)

- An Elgot algebra is an $H$-algebra $(A, \alpha : HA \to A)$ with an axiomatic iteration operator $(f : X \to A + HX) \mapsto (f^\dagger : X \to A)$ for $H$ being an arbitrary endofunctor

\(^5\)Hasuo, 2015, Generic weakest precondition semantics from monads enriched with order
Let us depict $X \rightarrow Y + HZ$ as

and thus adapt **Fixpoint** as

(merging feedforward wires amounts to calling $\alpha$)
Axiomatizing of Elgot Algebras: Uniformity

... Uniformity as

\[
\begin{align*}
Z & \xrightarrow{h} X \xrightarrow{f} A \\
X & \xrightarrow{\text{g}} Z \\
A & \xrightarrow{h} X
\end{align*}
\]

\[
\begin{align*}
Z & \xrightarrow{g} A \\
A & \xrightarrow{h} X
\end{align*}
\]
Naturality and Codiagonal cannot be adapted (no Bekić lemma for algebras!) and are replaced with

Compositionality:

\[ g \circ f \xrightarrow{\text{Compositionality}} g \]

---

6Adámek, Milius, and Velebil, 2006, Elgot Algebras
Naturality and Codiagonal cannot be adapted (no Bekić lemma for algebras!) and are replaced with

Compositionality:

Definition: an $H$-algebra $(A, \alpha)$ is a (complete) Elgot algebra if it is equipped with $(-)^\dagger$ satisfies Fixpoint, Uniformity and Compositionality.$^6$

$^6$Adámek, Milius, and Velebil, 2006, Elgot Algebras
Theorem:

1. Given an Elgot $H$-algebra $(A, \alpha, (-)^\dagger)$ and an $H$-algebra $(B, \beta)$, any iteration-congruent retraction $\rho : A \rightarrow B$ induces a canonical Elgot $H$-algebra structure on $B$.

2. Specifically, every Elgot $H$-algebra $(A, \alpha, (-)^\dagger)$ is obtained as an iteration-congruent retract of $(\nu \gamma \cdot A + H\gamma = H^\infty A, \mu, \ldots)$ under $(\text{out} : H^\infty A \rightarrow A + HH^\infty A)^\dagger$

(assuming that all the involved final coalgebras exist)
Theorem:

1. Given an Elgot $H$-algebra $(A, \alpha, (-)^\dagger)$ and an $H$-algebra $(B, \beta)$, any iteration-congruent retraction $\rho : A \to B$ induces a canonical Elgot $H$-algebra structure on $B$.

2. Specifically, every Elgot $H$-algebra $(A, \alpha, (-)^\dagger)$ is obtained as an iteration-congruent retract of $(\nu \gamma. A + H\gamma = H^\infty A, \mu, \ldots)$ under $(\text{out} : H^\infty A \to A + HH^\infty A)^\dagger$.

(assuming that all the involved final coalgebras exist)

Conjecture: A monad $T$ is an Elgot monad iff the free $T$-algebras $TX$ are Elgot $T$-algebras and the emerging retractions $T^\infty X \to TX$ jointly form a monad morphism.

(assuming all the involved final coalgebras exist)


Coalgebraic Methods in Computer Science.

Axioms for Iteration: Conway Operators

Let $T$ be a monad with a (total!) iteration operator $\_\_$. It is called a **Conway operator** if it additionally satisfies

**Dinaturality:**

$$
\begin{align*}
g(x, y, z) & = g(x, y, Z) \\
\end{align*}
$$

**Codiagonal:**

$$
\begin{align*}
g(x, y) & = g(x, x) \\
\end{align*}
$$