On Retracts of Algebras with Iteration

Sergey Goncharov[a]   Lutz Schröder[a]   Christoph Rauch[a]   Maciej Piróg[b]

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

[a]Friedrich-Alexander-Universität Erlangen-Nürnberg
[b]Uniwersytet Wrocławski

A Hierarchy of Structures for Iteration

- (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 $f : X \to TY$
  - 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 $(f : X \to TY) \mapsto (f^\star : TX \to TY)$

  inducing a category under

  $$\mathrm{id} = \eta : X \to TX \qquad f \diamond g = (f : Y \to TZ)^\star \circ (g : X \to TY)$$

  In Haskell's point-full notation: do $x \leftarrow p; f(x) = f^\star(p)$

Iteration for Monads

- An iteration operator assigns $f^\dagger : X \to TY$ to $f : X \to T(Y + X)$

Iteration for Monads

- An iteration operator assigns $f^\dagger : X \to TY$ to $f : X \to T(Y + X)$
- This yields a semantics for while loops:

$$\begin{aligned} &\mathrm{while}(b : X \to \mathrm{Bool}, p : X \to TX) \\ = &\big(\lambda x.\ \text{if } b(x) \text{ then } (T\,\mathrm{inr})(p(x)) \text{ else } (\eta\,\mathrm{inl})(x)\big)^\dagger \end{aligned}$$

(e.g. T is the partial store monad $(\text{–} \times S + 1)^S$)

Iteration for Monads

- An iteration operator assigns $f^\dagger : X \to TY$ to $f : X \to T(Y + X)$
- This yields a semantics for while loops:

$$\begin{aligned} &\mathsf{while}(b : X \to \mathsf{Bool}, p : X \to TX) \\ &= \big(\lambda x.\ \mathsf{if}\ b(x)\ \mathsf{then}\ (T\,\mathsf{inr})(p(x))\ \mathsf{else}\ (\eta\,\mathsf{inl})(x)\big)^\dagger \end{aligned}$$

(e.g. T is the partial store monad $(- \times S + 1)^S$)

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

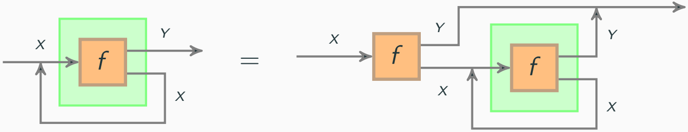
$$\begin{aligned} \mathsf{while}(b, p)(x) = \mathsf{if}\ b(x)\ \mathsf{then}\ &\mathsf{do}\ x \leftarrow p(x); \\ &\mathsf{while}(b, p)(x)\ \mathsf{else}\ \eta(x) \end{aligned}$$

can be couched in terms of $(-)^\dagger$

Elgot Monads (1/2)

A monad T is a (complete) Elgot monad[2] if it is equipped with an operator $(f : X \to T(Y + X)) \mapsto (f^\dagger : X \to TY)$ satisfying

Fixpoint:



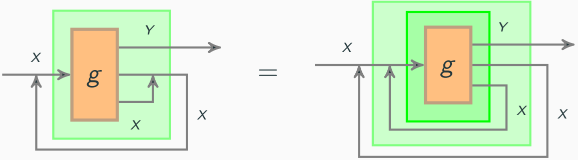
Naturality:

---

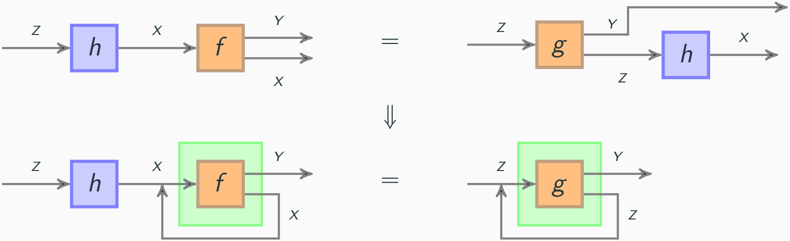
[2] Adámek, Milius, and Velebil, 2010, Equational properties of iterative monads

Elgot Monads (2/2)

Codiagonal:



Uniformity:



Iteration-Congruent Retractions

Our resent work[3] implies that Elgot monads are precisely characterized as iteration-congruent retracts:

$$\rho : \nu\gamma.\, \text{–} + T\gamma \;\leftrightarrows\; T : \upsilon$$

(ρ 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

Iteration-Congruent Retractions: Examples

Capretta's delay monad:

- In **Set**

$$\nu\gamma.\, X + \gamma \;\leftrightarrows\; X + 1$$

[4] Veltri 2017, A Type-Theoretical Study of Nontermination

Iteration-Congruent Retractions: Examples

Capretta's delay monad:

- In **Set**

$$\nu\gamma.\, X + \gamma \;\leftrightarrows\; X + 1$$

- In general, building

$$DX = \nu\gamma.\, X + \gamma \;\leftrightarrows\; \widetilde{D}X$$

  equalizing $\mathrm{id} : DX \to DX$ and $\mathrm{later} : DX \to DX$ 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 $(f : X \to 2 + \mathcal{P}X) \mapsto (f^{\ddagger} : X \to 2)$ 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 $\mathcal{P}$

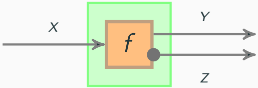
  This is relevant for weakest precondition semantics[5]

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

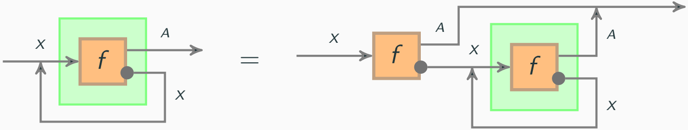
[5] Hasuo, 2015, Generic weakest precondition semantics from monads enriched with order

Axiomatizing of Elgot Algebras: Fixpoint

Let us depict $X \to Y + HZ$ as



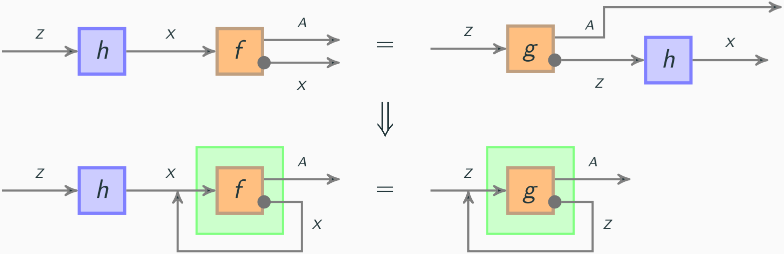
and thus adapt **Fixpoint** as



(merging feedforward wires amounts to calling α)

Axiomatizing of Elgot Algebras: Uniformity

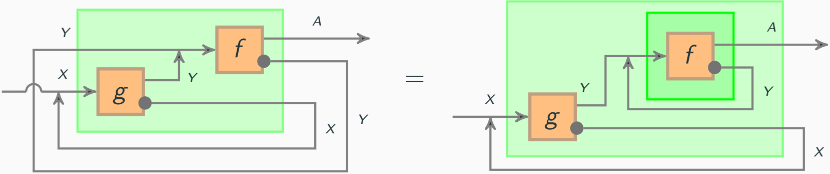
… **Uniformity** as



Axiomatizing of Elgot Algebras: Compositionality

Naturality and **Codiagonal** cannot be adapted (no Bekić lemma for algebras!) and are replaced with

Compositionality:

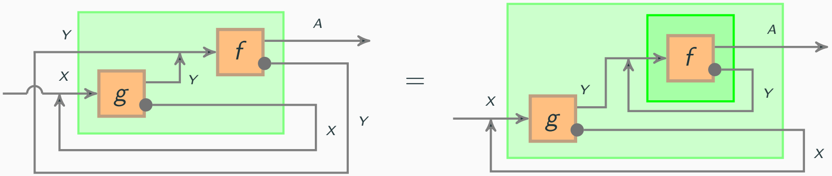


[6] Adámek, Milius, and Velebil, 2006, Elgot Algebras

Axiomatizing of Elgot Algebras: Compositionality

Naturality and **Codiagonal** cannot be adapted (no Bekić lemma for algebras!) and are replaced with

Compositionality:



Definition: an H-algebra (A, α) is a (complete) Elgot algebra if it is equipped with $(-)^{\ddagger}$ satisfies **Fixpoint**, **Uniformity** and **Compositionality**[6]

[6] Adámek, Milius, and Velebil, 2006, Elgot Algebras

Results

Theorem:

1. Given an Elgot H-algebra $(A, \alpha, (\text{–})^{\ddagger})$ and an H-algebra (B, β), any iteration-congruent retraction $\rho : A \to B$ induces a canonical Elgot H-algebra structure on B
2. Specifically, every Elgot H-algebra $(A, \alpha, (\text{–})^{\ddagger})$ is obtained as an iteration-congruent retract of $(\nu\gamma.\, A + H\gamma = H^{\infty}A, \mu, \ldots)$ under $(\mathsf{out} : H^{\infty}A \to A + HH^{\infty}A)^{\ddagger}$

   (assuming that all the involved final coalgebras exist)

Results

Theorem:

1. Given an Elgot H-algebra $(A, \alpha, (\text{–})^\ddagger)$ and an H-algebra (B, β), any iteration-congruent retraction $\rho : A \to B$ induces a canonical Elgot H-algebra structure on B
2. Specifically, every Elgot H-algebra $(A, \alpha, (\text{–})^\ddagger)$ is obtained as an iteration-congruent retract of $(\nu\gamma.\, A + H\gamma = H^\infty A, \mu, \ldots)$ under $(\mathsf{out} : H^\infty A \to A + HH^\infty A)^\ddagger$

   (assuming that all the involved final coalgebras exist)

Conjecture: A monad $\mathbb{T}$ is an Elgot monad iff the free $\mathbb{T}$-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)

References

Jiří Adámek, Stefan Milius, and Jiří Velebil. Elgot algebras. *Log. Methods Comput. Sci.*, 2(5:4):1–31, 2006.

Jiří Adámek, Stefan Milius, and Ji Velebil. Equational properties of iterative monads. *Inf. Comput.*, 208(12):1306–1348, 2010.

Sergey Goncharov and Lutz Schröder. Guarded traced categories. In Christel Baier and Ugo Dal Lago, editors, *Proc. 21th International Conference on Foundations of Software Science and Computation Structures (FoSSaCS 2018)*. Springer, 2018.

References II

Sergey Goncharov, Lutz Schröder, Christoph Rauch, and Maciej Piróg. Unifying guarded and unguarded iteration. In Javier Esparza and Andrzej Murawski, editors, *Foundations of Software Science and Computation Structures, FoSSaCS 2017*, volume 10203, pages 517–533. Springer, 2017.

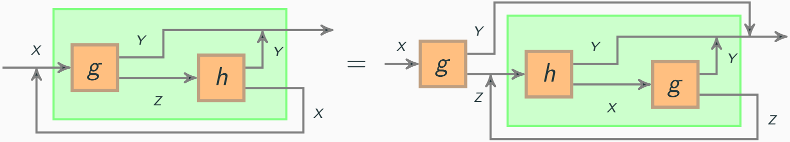
Ichiro Hasuo. Generic weakest precondition semantics from monads enriched with order. *Theoretical Computer Science*, 604:2 – 29, 2015. Coalgebraic Methods in Computer Science.

Niccoló Veltri. *A Type-Theoretical Study of Nontermination*. PhD thesis, Tallinn University of Technology, 2017.

Axioms for Iteration: Conway Operators

Let T be a monad with a (total!) iteration operator $-^{\dagger}$. It is called a Conway operator if it additionally satisfies

Dinaturality:



Codiagonal: