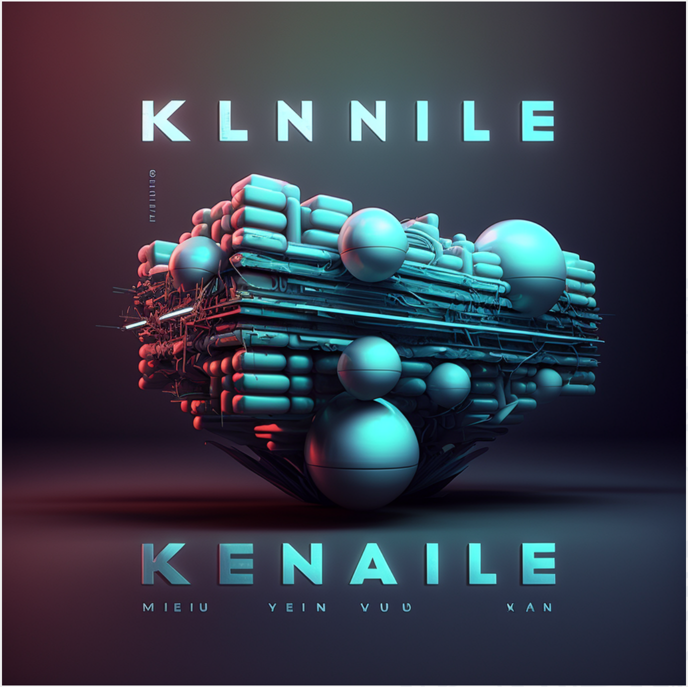

KLNNILE
KENAILE
MIEIU YEIN VUD XAN

Kleene Monads in a Long While

Sergey Goncharov

FAU Erlangen-Nürnberg

February 14, 2023, Oberseminar Theoretische Informatik

- Kleene iteration is iteration of regular expressions, e.g. $(0 \vee 1)^\star \cdot 0 \cdot (0 \vee 1)^\star$ – all binary strings that contain 0 at least once
- Kleene algebra is a lightweight equational theory of (Kleene) iteration, complete over formal languages
- It is extremely popular and has lots of extensions: hybrid, concurrent, stateful, etc
- Kleene monads is a simple categorification of Kleene algebras
- Elgot monads provide a vastly more general notion of iteration, and are highly compositional
  - But not quite that popular ☹
- What is precisely the delta between Kleene and Elgot?
- Can we bridge mathematical (mental, social, psychological, . . .) gap between them?

We approach by formalising a theory of while-loops

Plan

Recalling Kleene algebra

Recalling Kleene monads

Recalling Elgot monads

While-monads

Relating Kleene and Elgot

Conclusions

Kleene Algebra for (Kleene) Iteration

A Kleene algebra is a structure $(S, \bot, \eta, \vee, ;, (-)^*)$, where

- $(S, \bot, \eta, \vee, ;)$ is an idempotent semiring:
  - $(S, \bot, \vee)$ is a commutative ($x \vee y = y \vee x$) and idempotent ($x \vee x = x$) monoid
  - $(S, \eta, ;)$ is a monoid
  - distributive laws:

$$x; (y \vee z) = x; y \vee x; z \qquad x; \bot = \bot$$
$$(x \vee y); z = x; z \vee y; z \qquad \bot; x = \bot$$

  (thus, S is partially ordered: $x \leqslant y$ iff $x \vee y = y$)

- Kleene iteration satisfies $x^* = \eta \vee x; x^*$, and

$$\frac{x; y \vee z \leqslant y}{x^*; z \leqslant y} \qquad \frac{x \vee z; y \leqslant z}{x; y^* \leqslant z}$$

  Equivalently: $x^*; z$ is a least fixpoint of $x; (-) \vee z$ and $z; y^*$ is a least fixpoint of $(-); y \vee z$

Intuition: $\bot$ is a deadlock, η is a neutral program, $;$ is sequential composition, $\vee$ is non-deterministic choice

Use

- Regular expressions
- Algebraic language of finite state machines and beyond
- Relational semantics of programs
- Relational reasoning and verification, e.g. via dynamic logic
- Plenty of extensions:
  - modal $\Rightarrow$ modal Kleene algebra (Struth et al.)
  - stateful $\Rightarrow$ KAT + B! (Grathwohl, Kozen, Mamouras)
  - concurrent $\Rightarrow$ concurrent Kleene algebra (Hoare et al.)
  - nominal $\Rightarrow$ nominal Kleene algebra (Kozen et al.)
  - with differential equations $\Rightarrow$ differential dynamic logic (Platzer et al.)
  - etc., etc., etc.
- decidability and completeness properties (most famously w.r.t. formal languages and relational interpretations)

A minimalist extension is Kleene algebra with tests (KAT), which adds

- another Kleene algebra B of tests
- an operation-preserving inclusion $?: B \hookrightarrow S$
- complementation operator $\overline{(-)}: B \to B$, such that

$$\overline{a} \vee a = \top \qquad \overline{\overline{a}} = a \qquad \overline{a \vee b} = \overline{a}; \overline{b} \qquad \overline{\bot} = \eta$$

(this makes B into a Boolean algebra)

This enables encodings

Branching	(if b then p else q)	as	$b?; p \vee \overline{b}?; q$
Looping	(while b do p)	as	$(b?; p)^*; \overline{b}?$
Hoare triples	$\{a\}\, p\, \{b\}$	as	$a?; p; b? = a?; p$

In particular, we can embed deterministic semantics to non-deterministic semantics

Nondeterminism is a computational effect. If we want to add other effects, we need to add more operations and equations

Example: Probabilistic choice $(+_p \colon S \to S)_{p\in[0,1]}$, plus axioms of barycentric algebras

$$x +_0 y = y \qquad x +_p x = x \qquad x +_p y = x +_{1-p} y$$

$$(x +_p y) +_q z = x +_{\frac{p}{p+q-pq}} (y +_{p+q-pq} z)$$

plus distributivity $(x \vee y) +_p z = (x +_p z) \vee (y +_p z)$

Problem 1: Not all effects mix well with Kleene algebra, e.g. exception raising:

$$\mathsf{raise}\ \mathsf{e}_1 = \mathsf{raise}\ \mathsf{e}_1; \bot = \mathsf{raise}\ \mathsf{e}_2; \bot = \mathsf{raise}\ \mathsf{e}_2$$

Problem 2: We might want a stronger theory of non-determinism, e.g. the law $x; (y \vee z) = x; y \vee x; z$ is unsound w.r.t. strong bisimilarity

Problem 3: What is a disciplined way to extend iteration to other settings (muti-type, multi-effect, foundation-independent)?

Categorifying Iteration

Definition (Monad)

A monad **T** (on a category $\mathbf{C}$) is given by a Kleisli triple $(T, \eta, \text{–}^\star)$ where

- $T\colon |\mathbf{C}| \to |\mathbf{C}|$
- η is a family of morphisms $\eta_X : X \to TX$, forming monad unit
- $(\text{–})^\star$ assigns to each $f\colon X \to TY$ a morphism $f^\star : TX \to TY$

satisfying the laws: $\eta^\star = \mathrm{id}$, $f^\star\,\eta = f$, $(f^\star\,g)^\star = f^\star\,g^\star$

This entails that

- T is a functor, η is a natural transformation
- $f\colon X \to TY$ and $g\colon Y \to TZ$ (regarded as programs) can be Kleisli composed to $f;g = g \cdot f = g^\star f\colon X \to TZ$

By varying **T** we obtain various 'generalized programs' $f\colon X \to TY$ while programs of the form ηf can be seen as 'pure programs'

Example: $T =$ powerset $\Rightarrow$ generalized programs = non-deterministic programs, pure programs = deterministic programs = functions

Definition (Kleene-Kozen Category)

Call $\mathbf{C}$ a Kleene-Kozen category if it is enriched over join-semilattices with $\bot$ and strict join-preserving morphisms, and there is Kleene iteration

$$(\text{–})^* \colon \mathsf{Hom}(X, X) \to \mathsf{Hom}(X, X),$$

such that, given $f \colon Y \to Y$, $g \colon Y \to Z$ and $h \colon X \to Y$, gf^* is the least fixpoint of $g \vee (\text{–})f$ and f^*h is the least fixpoint of $h \vee f(\text{–})$

Proposition: A Kleene algebra is a Kleene-Kozen single-object category

Definition:[1] A monad $\mathbf{T}$ is a Kleene monad if its Kleisli category is Kleene-Kozen

This suggests a canonical definition of an n-sorted Kleene algebra as a Kleene-Kozen category with n objects

E.g. KAT becomes a certain two-sorted category

[1] Goncharov, Schröder, and Mossakowski, "Kleene monads: handling iteration in a framework of generic effects", 2009.

Examples:

- For any monoid M, $\mathcal{P}(M \times -)$ is a Kleene monad (generalizes language-theoretic models)
- If T is a Kleene monad then so is the state-monad transform $(T(- \times S))^S$ for any S (generalized relational models)

Non-Examples:

- Maybe monad $TX = X + 1$ has $\bot$ but no $\vee$
- $TX = \mathcal{P}^+(X+1)$ (where $\mathcal{P}^+$ is non-empty powerset) fails $\boxed{f \vee \bot = f}$
- $TX = \mathcal{P}(A^\star \times X + A^\omega)$ fails $\boxed{f; \bot = \bot}$
- TX is a final coalgebra for $\mathcal{P}_{\omega_1}(X + A \times (-))$ (where $\mathcal{P}_{\omega_1}$ is the countable powerset functor) fails $\boxed{f;(g \vee h) = f;g \vee f;h}$

Definition (Elgot monad)

A (complete) Elgot monad[2] in a category with binary coproducts is a monad **T** equipped with an Elgot iteration operator

$$(–)^{\dagger}\colon \mathsf{Hom}(X, T(Y+X)) \to \mathsf{Hom}(X, TY),$$

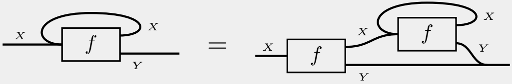
satisfying four laws: fixpoint, uniformily, naturality and codiagonal

- Elgotness is robust and stable under many monad transformers
  - $T \mapsto T(M \times –)$ (writer)
  - $T \mapsto T(– + E)$ (exception)
  - $T \mapsto (T(– \times S))^S$ (state)
  - $T \mapsto \nu\gamma.\, T(– + H\gamma)$ (resumption)
- Laws go back to Elgot[3], except for uniformity
- All previous examples are Elgot

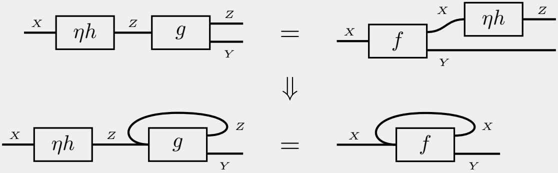
[2] Adámek, Milius, and Velebil, "Equational properties of iterative monads", 2010.
[3] Elgot, "Monadic Computation And Iterative Algebraic Theories", 1975.

Fixpoint and Uniformity

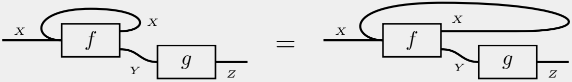
Fixpoint ($f\colon X \to T(Y+X)$)**:**



Uniformity ($f\colon X \to T(Z+X)$, $g\colon Y \to T(Z+Y)$, $h\colon X \to Y$)**:**



Naturality ($f\colon X \to T(Y+X)$, $g\colon Y \to TZ$):

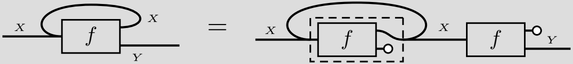


Codiagonal ($f\colon X \to T(Y+(X+X))$):

Naturality and **Codiagonal** are basically coherence laws

Theorem

Every Kleene monad is an Elgot monad under



where —o is the deadlock

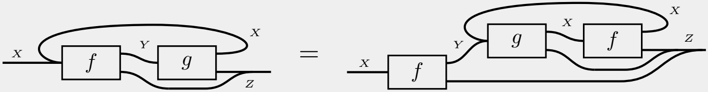
Proof.

Essentially a known fact about iteration theories[4] □

But, how large is the gap between Kleene monads and Elgot monads + $\{\bot, \vee\}$?

[4] Cazanescu and Stefanescu, "Feedback, Iteration and Repetition", 1994.

The **Dinaturality** law



is derivable[5], also it was included in the original axiomatization

Q: So, maybe there are more derivable axioms?

A: In fact, it is provable that the present axiomatization is minimal

[5] Ésik and Goncharov, "Some Remarks on Conway and Iteration Theories", 2016.

While-Monads

- A key distinction between Elgot iteration and Kleene iteration is that the former needs a (simple) type system, while the later can make do without any types whatsoever
- Kleene algebra with tests has two sorts, and yet no types

Definition (Decisions)

We call a family $(\mathbf{C}^{\mathsf{d}}(X) \subseteq \mathbf{C}(X, X + X))_{X \in |\mathbf{C}|}$ in a category $\mathbf{C}$, decisions if every $\mathbf{C}^{\mathsf{d}}(X)$ contains inl, inr, and $\mathbf{C}^{\mathsf{d}}$ is closed under if-then-else.

We encode logical operations on decisions as follows:

$$\mathsf{ff} = \mathsf{inl}, \qquad b \mathbin{\&\&} c = \mathsf{if}\ b\ \mathsf{then}\ c\ \mathsf{else}\ \mathsf{ff}, \qquad {\sim} b = \mathsf{if}\ b\ \mathsf{then}\ \mathsf{ff}\ \mathsf{else}\ \mathsf{tt},$$
$$\mathsf{ff} = \mathsf{inr}, \qquad b \mathbin{||} c = \mathsf{if}\ b\ \mathsf{then}\ \mathsf{tt}\ \mathsf{else}\ c.$$

By definition, decisions can range from the smallest family with $\mathbf{C}^{\mathsf{d}}(X) = \{\mathsf{ff}, \mathsf{tt}\}$, to the greatest one with $\mathbf{C}^{\mathsf{d}}(X) = \mathbf{C}(X, X + X)$

A while-monad is a monad **T**, equipped with an operator

$$\mathsf{while}\colon \mathbf{C}^{\mathsf{d}}_{\mathsf{T}}(X) \times \mathbf{C}(X, TX) \to \mathbf{C}(X, TX),$$

such that the following axioms are satisfied

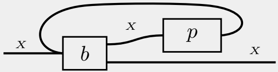
W-Fix $\mathsf{while}\ b\ p = \mathsf{if}\ b\ \mathsf{then}\ p; (\mathsf{while}\ b\ p)\ \mathsf{else}\ \eta$

W-Or $\mathsf{while}\,(b\ ||\ c)\ p = (\mathsf{while}\ b\ p); \mathsf{while}\ c\ (p; \mathsf{while}\ b\ p)$

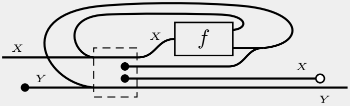
W-And $$\frac{\eta h; b = \eta u; \mathsf{ff}}{\mathsf{while}\,(b\ \&\&\ (c\ ||\ \eta u; \mathsf{ff}))\ p = \mathsf{while}\ b\ (\mathsf{if}\ c\ \mathsf{then}\ p\ \mathsf{else}\ \eta h)}$$

W-Uni $$\frac{\eta h; b = \mathsf{if}\ c\ \mathsf{then}\ \eta h'; \mathsf{tt}\ \mathsf{else}\ \eta u; \mathsf{ff} \qquad \eta h'; p = q; \eta h}{\eta h; \mathsf{while}\ b\ p = (\mathsf{while}\ c\ q); \eta u}$$

- From $(–)^\dagger$ to while : while $b\ p = (\text{if } b \text{ then } p; \text{tt else ff})^\dagger$, diagrammatically, while $b\ p$ is expressed as



- From while to $(–)^\dagger$



Theorem

If for all $X, Y \in |\mathbf{C}|$, $\eta(\mathsf{inl} + \mathsf{inr}) \in \mathbf{C}^{\mathsf{d}}_{\mathsf{T}}(X + Y)$ then T is and Elgot monad iff it is a while-monad w.r.t. $\mathbf{C}^{\mathsf{d}}_{\mathsf{T}}$

Theorem

A monad **T** *is a Kleene monad iff*

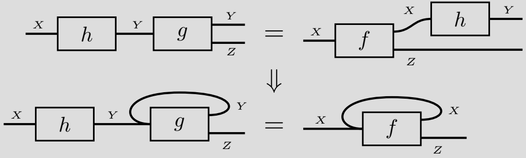
- *the Kleisli category of* **T** *is enriched over join-semilattices with* $\bot$ *and strict join-preserving morphisms;*
- *there is an operator* $(-)^*\colon \mathsf{Hom}(X, TX) \to \mathsf{Hom}(X, TX)$*, such that*
  1. $f^* = \eta \vee f^* \cdot f$
  2. $\eta^* = \eta$
  3. $f^* = (f \vee \eta)^*$
  4. $f \cdot h = g \cdot f \;\Rightarrow\; f \cdot h^* = g^* \cdot f$

Corollary: Since Kleene algebra is a special case, this is also a complete axiomatization for Kleene algebra

Theorem

A monad **T** *is a Kleene monad iff*

1. **T** *is Elgot*
2. *the Kleisli category of* **T** *is enriched over bounded join-semilattices and strict join-preserving morphisms*
3. **T** *satisfies the law* $(\eta\,\mathsf{inl} \vee \eta\,\mathsf{inr})^\dagger = \eta$
4. **T** *satisfies strong uniformity:*



where h *is strict* $h;\delta = \delta$*, and where* $\delta = (\eta\,\mathsf{inr})^\dagger$ *by definition*

- Elgot monads that fail strong uniformity without the strictness assumption on h are easy to manufacture
  - For example, $\mathcal{P}(\text{–} + 1)$ is an Elgot monad, obtained by transforming $\mathcal{P}$
  - Besides $\delta = \{\mathsf{inr}\,\star\}$ it contains another 'divergence' $\bot = \{\,\}$
  - The premise of strong uniformity is satisfied with $h = \bot$ and $f = \eta\,\mathsf{inr}$, but the conclusion $\delta = (\eta\,\mathsf{inr})^\dagger = g^\dagger$ generally fails
- It is thus always reasonable to restrict h in strong uniformity to strict programs, i.e. $h; \delta = \delta$
- If we add finite nondeterminism and other laws, it is more difficult to construct a separating example, however

Proposition

There exists an Elgot monad **T**, whose Kleisli category is enriched over bounded semi-lattices, $(\eta\,\mathsf{inl} \vee \eta\,\mathsf{inr})^\dagger = \eta$, but **T** fails strong uniformity.

1. The idea is based on Kozen's separating example for left-handed and right-handed Kleene algebras[6]
2. Consider the submonad **T** of the continuation monad (neighbourhood monad, dualization monad) $(\text{–} \to 2) \to 2$, formed by those f that preserve finite unions:

$$f\{\,\} = \{\,\} \qquad\qquad f(A \cup B) = f(A) \cup f(B)$$

3. Using the fact that every TX is a complete lattice, define $f^\dagger$ as a least fixpoint, using the Knaster-Tarski theorem. Hence $(\eta\, \mathsf{inl} \vee \eta\, \mathsf{inr})^\dagger = \eta$
4. Enrichment in semilattices follow by definition
5. Remaining Elgot monad laws follow by transfinite induction
6. If **T** was a Kleene monad, any $\mathsf{Hom}(X, TX)$ would be a Kleene algebra, but Kozen showed it is not $\Rightarrow$ **T** is not Kleene

[6] Kozen, "On Kleene Algebras and Closed Semirings", 1990.

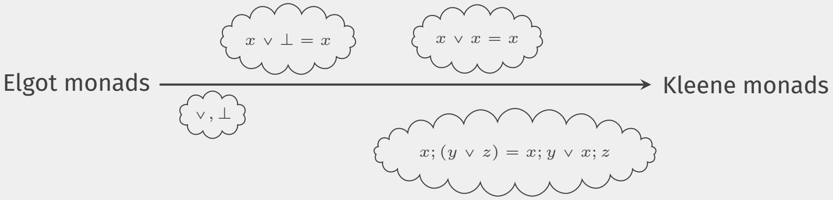
So, what is the fundamental version of uniformity?

- Bloom and Ésik[7] argued that **there is only one theory of iteration**, accepting only those instances of uniformity, where premises are internally provable. This is insufficient for practical equational reasoning
- Uniformity is arguable the most conservative extension of the purely equational theory and is a part of the Elgot monad axiomatization
- Strong uniformity with strict maps bridges the gap with Kleene monads

[7] Bloom and Ésik, *Iteration theories: the equational logic of iterative processes*, 1993.

Conclusions

We obtained a spectrum



Further Work:

- How can we define while-algebras, generalizing KAT, in the upshot?
  - The main hurdle is uniformity, which should preferably not allude to a yet another sort of programs, and be properly weaker than strong uniformity
- When can we equivalently replace while $+ \vee$ with Kleene star $+ \vee$ so that $p^* = \text{while}\,(\text{ff} \vee \text{tt})\ p$?
- Generic completeness theorems, generalizing completeness for KAT