Unifying Categorical View of Nondeterministic Iteration and Tests

Sergey Goncharov and Tarmo Uustalu

FAU Erlangen-Nürnberg

Reykjavik University
Tallinn University of Technology

CONCUR 2024, 9–13 September 2024

Plan

Recap of Kleene algebra

Lack of Compositionality/Robustness

Uniform Conway Iteration

Kleene-Iteration Categories (with Tests)

Conclusions

Kleene Algebra: Definition

Kleene algebra is

- idempotent semiring $(S, 0, 1, +, ;)$
  - $(S, 0, +)$ is commutative and idempotent monoid
  - $(S, 1, ;)$ is monoid
  - distributive laws:

$$p; (q + r) = p; q + p; r \qquad\qquad p; 0 = 0$$
$$(p + q); r = p; r + q; r \qquad\qquad 0; p = 0$$

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

- ... plus, Kleene iteration, satisfying $p^* = 1 + p; p^*$, and

$$\frac{p; r + q \leqslant r}{p^*; q \leqslant r} \qquad\qquad \frac{r; p + q \leqslant r}{q; p^* \leqslant r}$$

$\Rightarrow$ $p^*; q$ and $p; q^*$ are least (pre-)fixpoints

Key Design Features

- Not tailored to language model – complete also over relational model
- Algebraic: closed under substitution, in contrast to Salomaa's rule*

$$\frac{r = pr + q \qquad p \text{ guarded}}{r = p^* q}$$

- All fixpoints are least (pre-)fixpoints
  - in Salomaa's system: particular fixpoints are unique fixpoints
- Induction rules

$$\frac{p; r + q \leqslant r}{p^*; q \leqslant r} \qquad\qquad \frac{r; p + q \leqslant r}{q; p^* \leqslant r}$$

  encompass infinitely many identities, critical for completeness
- Completeness via free model = regular languages (regular events)

*A. Salomaa, Two Complete Axiom Systems for the Algebra of Regular Events, 1966

Kleene Algebra: 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.)
  - differential equations $\Rightarrow$ differential dynamic logic (Platzer et al.)
  - network primitives $\Rightarrow$ NetKAT (Foster et al.)
  - etc., etc., etc.
- decidability and completeness (most famously w.r.t. language interpretation and relational interpretation)

Tests for Control

- Programming view: algebra elements = programs
  - 0 – divergence and/or deadlock, 1 – neutral program, etc.
- Kleene algebra with tests (KAT) adds control via tests:
  - Kleene sub-algebra B
  - B is Boolean algebra under $(0, 1, ;, +)$
- This enables encodings:

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

Example:

$$\text{while } b \text{ do } p = \text{if } b \text{ then } p \text{ else } (\text{while } b \text{ do } p)$$

Lack of Compositionality /Robustness

Scenario I: Exceptions

- Assumming programs raise exceptions: $\mathrm{raise}\, e_i =$ "raise exception e_i",

$$\mathrm{raise}\, e_1 = \mathrm{raise}\, e_1 ; 0 = 0 = \mathrm{raise}\, e_2 ; 0 = \mathrm{raise}\, e_2$$

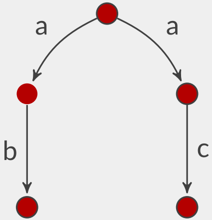
- So, we cannot have more than one exception

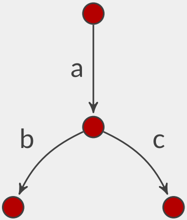
  ... unless we discard the law

$$p ; 0 = 0$$

Scenario II: Branching Time

Processes





are famously non-bisimular, failing Kleene algebra law

$$p; (q + r) = p; q + p; r$$

Scenario III: Divergence

- Identity

$$(p+1)^* = p^*$$

  is provable in Kleene algebra, because p^* is a least fixpoint

- Alternatively:

$$1^* = 1$$

- Hence deadlock = divergence

? How to undo this

What is generic core of Kleene iteration?

- Core reasoning principles
- Robustness under adding features (e.g. exceptions)
- Generic completeness argument
- Compliance with classical program semantics

What is generic core of Kleene iteration?

- Core reasoning principles
- Robustness under adding features (e.g. exceptions)
- Generic completeness argument
- Compliance with classical program semantics

Categorifying Iteration

From Algebras to Categories

- Categories $\approx$ many-sorted monoids:

$$1_A\colon A \to A \quad \text{(unit)} \qquad \frac{p\colon A \to B \qquad q\colon B \to C}{p;q\colon A \to C} \quad \text{(multiplication)}$$

  - Objects $A, B, \ldots$ – sorts, Morphisms $p\colon A \to B$ – programs
  - **Fact:** monoid = single-object category

- Kleene-Kozen categories – additionaly

$$0_{A,B}\colon A \to B \qquad \frac{p\colon A \to B \qquad q\colon A \to B}{p+q\colon A \to B} \qquad \frac{p\colon A \to A}{p^*\colon A \to A}$$

  subject to Kleene algebra laws

  - **Fact:** Kleene algebra = single-object Kleene-Kozen category
  - **Example:** Category of relations = relational interpretation

- Tests = particular morphisms $b\colon A \to A$ forming Boolean algebra

Uniformity for Kleene Iteration

- Alternative Kleene algebra axiomatization*: idempotent semiring, plus

  Axioms:

  $$p^* = 1 + p; p^* \qquad (p+q)^* = p^*; (q; p^*)^* \qquad 1^* = 1$$

  Uniformity rule:

  $$\frac{u; p = q; u}{u; p^* = q^*; u}$$

- Same for Kleene-Kozen categories
- Removing $1^* = 1$ yields "may-diverge Kleene algebras"
  $\Rightarrow (-)^*$ is no longer least fixpoint (!)
- Uniformity is assumed for arbitrary u

*S. Goncharov, Shades of Iteration: From Elgot to Kleene, 2023

Uniformity for Kleene Iteration

- Alternative Kleene algebra axiomatization*: idempotent semiring, plus

  Axioms:

  $$p^* = 1 + p; p^* \qquad (p+q)^* = p^*; (q; p^*)^* \qquad 1^* = 1$$

  Uniformity rule:

  $$\frac{u; p = q; u}{u; p^* = q^*; u}$$

- Same for Kleene-Kozen categories
- Removing $1^* = 1$ yields "may-diverge Kleene algebras"
  $\Rightarrow (-)^*$ is no longer least fixpoint (!)
- Uniformity is assumed for arbitrary u **!**

*S. Goncharov, Shades of Iteration: From Elgot to Kleene, 2023

Coproducts and Elgot Iteration

- Coproducts $A \oplus B$ generalize disjoint unions $A \uplus B$
- Elgot iteration:

$$\frac{p \colon A \to B \oplus A}{p^\dagger \colon A \to B}$$

  - Intuitively: keep running p until reached a result in B
- Uniform Conway iteration additionally satisfies standard equational laws and uniformity principle
- $(-)^\dagger$ is subject to rich and elaborated equational theory of iteration*

☺ Very general
☺ Robust under adding features (states, reading, writing, exceptions, process algebra actions)
☺ Does not hinge on non-determinism
☹ Hinges on coproducts

*S. Bloom, **Z. Ésik**, Iteration Theories, 1993

Restricting Uniformity

Taking $u = \text{raise } e$ in

$$\frac{u; p = q; u}{u; p^* = q^*; u}$$

allows for producing

$$\text{raise } e = 1^*; \text{raise } e$$

❗ Need not hold in may-diverge Kleene algebras
$\Rightarrow$ Restrict to linear u:

$$u; 0 = 0 \qquad u; (p + q) = u; p + u; q$$

Kleene-iteration category with tests (KiCT):

- Category with coproducts and nondeterminism
- Selected class of tests
- Selected class of linear tame morphisms
- Kleene iteration
- Laws:

$$0; p = 0 \qquad (p+q); r = p; r + q; r$$

$$p^* = 1 + p; p^* \qquad (p+q)^* = p^*; (q; p^*)^*$$

$$\frac{u; p^* = q^*; u}{u; p = q; u}$$

for tame u

Properties

- KiCT + $(1^* = 1)$ with all morphisms tame = Kleene-Kozen with tests and coproducts
- KiCT with expressive tests = tame-uniform Conway iteration + non-determinism (“expressive” = $A \oplus B \xrightarrow{[\mathsf{inl},\mathsf{o}]} A \oplus B$ are included)
- Free KiCT = non-deterministic rational trees w.r.t. may-diverge nondeterminism

Glimpse into Completeness

- Completeness is obtained by characterising free KiCT
- Morphisms of free KiCT are forests of infinite rational strongly extensional associative-commutative-idempotent trees:

$$\mathfrak{T} = \sum_{i \in I} b_i.\, \mathsf{u}_i.\, \mathfrak{T}_i + \sum_{i \in J} b_i.\, \mathsf{f}_i(\mathfrak{T}_{i,1}, \ldots, \mathfrak{T}_{i,n_i}) + \sum_{i \in K} b_i.\, x_i$$

- I, J, K are countable and disjoint
- b_i range over guarded strings $c_1?u_1c_2?\ldots u_nc_{n+1}?$ ($c_k?$ – tests, u_k – tame morphisms)
- u_i range over tame morphisms
- f_i range over non-tame morphisms

- Rationality = finitely many distinct subtrees
- Strong extensionality = bisimular subtrees are equal

What is generic core of Kleene iteration?

KiCT:

- ✅ Core reasoning principles
- ✅ Robustness under adding features
- ✅ Generic completeness argument
- ✅ Compatibility with classical program semantics

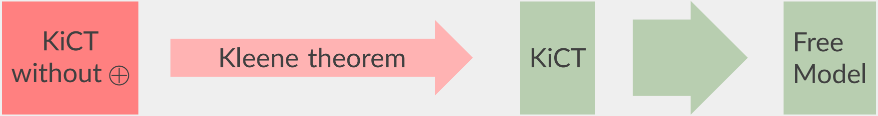
What is generic core of Kleene iteration?

KiCT:

- ✓ Core reasoning principles
- ✓ Robustness under adding features
- ✓ Generic completeness argument
- ✓ Compatibility with classical program semantics

But what is KiCT without coproducts?

Hypothetical Route



- If everything is tame (Kleene algebra), this is essentially what happens
- This route also works for may-diverge Kleene algebras (Conjecture)
- What if nothing is tame (Process algebra)?

Milner's Conundrum

- Milner* realized under strong bisimilarity that "regular behaviours" are properly more general than "$*$-behaviours"
- Simplest example

$$\begin{cases} X = \mathbf{1} + a; Y \\ Y = \mathbf{1} + b; X \end{cases}$$

We can pass to $X = \mathbf{1} + a; (\mathbf{1} + b; X)$, but not to $X = (ab)^*(\mathbf{1} + a)$

- This descrepancy $\approx$ failure of Kleene theorem
- Milner's solution $\approx$ using coproducts in the language
- He also proposed a modification of Salomaa's system for $*$-behaviours – proven complete only recently (Grabmayer)

*R. Milner, A complete inference system for a class of regular behaviours, 1984

Conclusions

- KiCTs reframe Kleene algebra principles in categorical setting and succeed with various yardsticks
- KiCTs without coproducts would be a hypothetical most basic notions of Kleene iteration
- **Open Problem:** Can it ever be found?

Compositionality and Robustness

- Given Elgot iteration operator, fix carrier of exceptions E
- Exception-raising morphisms $A \to B \oplus E$ themeselves form a category
- Elgot iteration and its laws carry over
  - This fails for Kleene-Kozen categories
- Elgot iteration's laws are thus stable under exception monad transformer
- Similarly: state, reading, writing, adjoining process algebra actions

While-Iteration

- Let tests be selected morphisms $b, c, \ldots : A \to A$, forming Boolean algebra under $1, 0, ;, +$
- While-iteration:

Axioms:

$$\text{while } b \text{ do } p = \text{if } b \text{ then } p; (\text{while } b \text{ do } p) \text{ else } 1$$

$$\text{while } (b + c) \text{ do } p = (\text{while } b \text{ do } p); \text{while } c \text{ do } (p; \text{while } b \text{ do } p)$$

Uniformity:

$$\frac{u; b; p = c; q; u \qquad u; \bar{b} = \bar{c}; v}{u; \text{while } b \text{ do } p = (\text{while } c \text{ do } q); v}$$

Theorem: Non-deterministic uniform Conway iteration is equivalent to while-iteration, provided tests contain all $A \oplus B \xrightarrow{[\mathrm{inl}, 0]} A \oplus B$