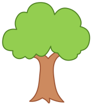
Distributive Laws in the Boom Hierarchy

Maaike Zwart

University of Oxford

2 December 2020



DEPARTMENT OF COMPUTER SCIENCE

UNIVERSITY OF OXFORD

Overview

- Introduction
  - Motivation: monads and monad compositions
  - Reminder: algebraic theories and composites
  - My strategy proving for no-go theorems
- Boom hierarchy: examples and intuition
- Spotlight theorem: too many constants theorem
- Conclusion

Motivation: monads and monad compositions

A monad is a categorical structure used for:

- Modelling of data structures (lists, trees, etc)

Motivation: monads and monad compositions

A monad is a categorical structure used for:

- Modelling of data structures (lists, trees, etc)
- Modelling of computation (exception, reader, writer, etc)

Motivation: monads and monad compositions

A monad is a categorical structure used for:

- Modelling of data structures (lists, trees, etc)
- Modelling of computation (exception, reader, writer, etc)

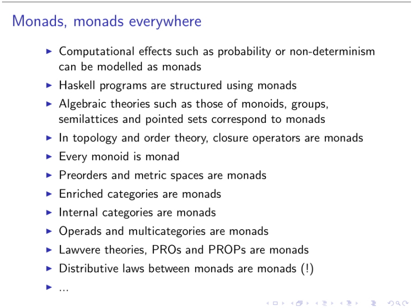
Monads, monads everywhere

- Computational effects such as probability or non-determinism can be modelled as monads
- Haskell programs are structured using monads
- Algebraic theories such as those of monoids, groups, semilattices and pointed sets correspond to monads
- In topology and order theory, closure operators are monads
- Every monoid is monad
- Preorders and metric spaces are monads
- Enriched categories are monads
- Internal categories are monads
- Operads and multicategories are monads
- Lawvere theories, PROs and PROPs are monads
- Distributive laws between monads are monads (!)
- ...

Motivation: monads and monad compositions

A monad is a categorical structure used for:

- Modelling of data structures (lists, trees, etc)
- Modelling of computation (exception, reader, writer, etc)



Monads, monads everywhere

- Computational effects such as probability or non-determinism can be modelled as monads
- Haskell programs are structured using monads
- Algebraic theories such as those of monoids, groups, semilattices and pointed sets correspond to monads
- In topology and order theory, closure operators are monads
- Every monoid is monad
- Preorders and metric spaces are monads
- Enriched categories are monads
- Internal categories are monads
- Operads and multicategories are monads
- Lawvere theories, PROs and PROPs are monads
- Distributive laws between monads are monads (!)
- ...

Compositions of monads allow simultaneous modelling of multiple computational aspects.

Monads: What are they?

A monad is a triple $\langle T, \eta, \mu \rangle$, with T an endofunctor and $\eta : 1 \Rightarrow T$, $\mu : TT \Rightarrow T$ natural transformations, such that:

$$
\begin{array}{ccc}
T & \xrightarrow{\eta T} & TT \\
{\scriptstyle T\eta}\downarrow & \searrow^{Id} & \downarrow{\scriptstyle \mu} \\
TT & \xrightarrow{\mu} & T
\end{array}
\qquad
\begin{array}{ccc}
TTT & \xrightarrow{T\mu} & TT \\
{\scriptstyle \mu T}\downarrow & & \downarrow{\scriptstyle \mu} \\
TT & \xrightarrow{\mu} & T
\end{array}
$$

Monads: What are they?

A monad is a triple $\langle T, \eta, \mu \rangle$, with T an endofunctor and $\eta : 1 \Rightarrow T$, $\mu : TT \Rightarrow T$ natural transformations, such that:

$$
\begin{array}{ccc}
T & \xrightarrow{\eta T} & TT \\
{\scriptstyle T\eta}\downarrow & \searrow^{Id} & \downarrow{\scriptstyle \mu} \\
TT & \xrightarrow{\mu} & T
\end{array}
\qquad\qquad
\begin{array}{ccc}
TTT & \xrightarrow{T\mu} & TT \\
{\scriptstyle \mu T}\downarrow & & \downarrow{\scriptstyle \mu} \\
TT & \xrightarrow{\mu} & T
\end{array}
$$

Examples:

- List
- Multiset/Bag
- Powerset
- Distribution
- Exception
- Writer
- Reader

Composing Monads

- Find η^{TS}, μ^{TS} such that $\langle TS, \eta^{TS}, \mu^{TS} \rangle$ is a monad.

Composing Monads

- Find η^{TS}, μ^{TS} such that $\langle TS, \eta^{TS}, \mu^{TS} \rangle$ is a monad.
- Good candidate for η^{TS}:

$$\eta^T \eta^S : 1 \Rightarrow TS$$

Composing Monads

- Find η^{TS}, μ^{TS} such that $\langle TS, \eta^{TS}, \mu^{TS} \rangle$ is a monad.
- Good candidate for η^{TS}:

$$\eta^T \eta^S : 1 \Rightarrow TS$$

- Same for μ^{TS}?

Composing Monads

- Find η^{TS}, μ^{TS} such that $\langle TS, \eta^{TS}, \mu^{TS} \rangle$ is a monad.
- Good candidate for η^{TS}:

$$\eta^T \eta^S : 1 \Rightarrow TS$$

- Same for μ^{TS}?
  - Need:

$$\mu^{TS} : TSTS \Rightarrow TS$$

Composing Monads

- Find η^{TS}, μ^{TS} such that $\langle TS, \eta^{TS}, \mu^{TS} \rangle$ is a monad.
- Good candidate for η^{TS}:

$$\eta^T \eta^S : 1 \Rightarrow TS$$

- Same for μ^{TS}?
  - Need:

$$\mu^{TS} : TSTS \Rightarrow TS$$

  - Have:

$$\mu^T \mu^S : TTSS \Rightarrow TS$$

Composing Monads

- Find η^{TS}, μ^{TS} such that $\langle TS, \eta^{TS}, \mu^{TS} \rangle$ is a monad.
- Good candidate for η^{TS}:

$$\eta^T \eta^S : 1 \Rightarrow TS$$

- Same for μ^{TS}?
  - Need:

$$\mu^{TS} : TSTS \Rightarrow TS$$

  - Have:

$$\mu^T \mu^S : TTSS \Rightarrow TS$$

  - Solution:

$$\lambda : ST \Rightarrow TS$$

Composing Monads

- Find η^{TS}, μ^{TS} such that $\langle TS, \eta^{TS}, \mu^{TS} \rangle$ is a monad.
- Good candidate for η^{TS}:

$$\eta^T \eta^S : 1 \Rightarrow TS$$

- Same for μ^{TS}?
    - Need:

$$\mu^{TS} : TSTS \Rightarrow TS$$

    - Have:

$$\mu^T \mu^S : TTSS \Rightarrow TS$$

    - Solution:

$$\lambda : ST \Rightarrow TS$$

- If λ is a *distributive law*, then the above choices form a monad.

- Beck 1969.

Motivation: monads and distributive laws

Problem:

- Distributive laws are hard to find.
  (time consuming)

Motivation: monads and distributive laws

Problem:

- Distributive laws are hard to find.
  (time consuming)
- Axioms are hard to check.
  (resulting in mistakes in the literature!)

Motivation: monads and distributive laws

Problem:

- Distributive laws are hard to find.
  (time consuming)
- Axioms are hard to check.
  (resulting in mistakes in the literature!)
- Distributive laws might not even exist.

Motivation: monads and distributive laws

Problem:

- Distributive laws are hard to find.
  (time consuming)
- Axioms are hard to check.
  (resulting in mistakes in the literature!)
- Distributive laws might not even exist.

Motivation: monads and distributive laws

Problem:

- Distributive laws are hard to find.
  (time consuming)
- Axioms are hard to check.
  (resulting in mistakes in the literature!)
- Distributive laws might not even exist.

What I do:

- Find no-go theorems for distributive laws.

Motivation: monads and distributive laws

Problem:

- Distributive laws are hard to find.
  (time consuming)
- Axioms are hard to check.
  (resulting in mistakes in the literature!)
- Distributive laws might not even exist.

What I do:

- Find no-go theorems for distributive laws.

My weapon of choice:

- Algebra.

A quick reminder: algebraic theories

Algebraic theory:

- Signature Σ and a set of variables give *terms*.
- Axioms E and equational logic give equivalence of terms.

Reflexivity: $$\frac{}{t = t}$$

Axiom: $$\frac{(s, t) \in E}{s = t}$$

Symmetry: $$\frac{t = t'}{t' = t}$$

Substitution: $$\frac{t = t'}{t[f] = t'[f]}$$

Transitivity: $$\frac{t = t',\, t' = t''}{t' = t''}$$

For any σ: $$\frac{t_1 = t_1', \ldots, t_n = t_n'}{\sigma(t_1, \ldots, t_n) = \sigma(t_1', \ldots, t_n')}$$

A quick reminder: algebraic theories

Algebraic theory:

- Signature Σ and a set of variables give *terms*.
- Axioms E and equational logic give equivalence of terms.

Monoids:

$$\Sigma = \{1^{(0)}, *^{(2)}\}$$
$$E = \{1 * x = x = x * 1, \quad (x * y) * z = x * (y * z)\}$$

Abelian groups:

$$\Sigma = \{0^{(0)}, -^{(1)}, +^{(2)}\}$$
$$E = \{0 + x = x = x + 0, \quad (x + y) + z = x + (y + z), \quad x + y = y + x, \quad x + (-x) = 0 = (-x) + x\}$$

A quick reminder: algebraic theories

Algebraic theory:

- Signature Σ and a set of variables give *terms*.
- Axioms E and equational logic give equivalence of terms.

Monoids:

$$\Sigma = \{1^{(0)}, *^{(2)}\}$$
$$E = \{1 * x = x = x * 1, \quad (x * y) * z = x * (y * z)\}$$

Abelian groups:

$$\Sigma = \{0^{(0)}, -^{(1)}, +^{(2)}\}$$
$$E = \{0 + x = x = x + 0, \quad (x + y) + z = x + (y + z), \quad x + y = y + x, \quad x + (-x) = 0 = (-x) + x\}$$

Monads arise from free/forgetful adjunction between Set and category of (Σ, E)-algebras.

Composite theories: the equivalent of distributive laws

Example: Rings are a *composite theory*[1] of Abelian groups after Monoids.

Rings:

$$
\begin{aligned}
\Sigma &= \Sigma^A \uplus \Sigma^M \\
&= \{0^{(0)}, 1^{(0)}, -^{(1)}, +^{(2)}, *^{(2)}\} \\
E &= E^A \cup E^M \cup \\
&\quad \{a * (b + c) = (a * b) + (a * c) \\
&\quad (a + b) * c = (a * c) + (b * c)\}
\end{aligned}
$$

[1] Piróg and Staton 2017.

My strategy: no-go theorems for distributive laws

Using composite theories:

- Choose two theories to compose.

My strategy: no-go theorems for distributive laws

Using composite theories:

- Choose two theories to compose.
- Assume composite theory exists.

My strategy: no-go theorems for distributive laws

Using composite theories:

- Choose two theories to compose.
- Assume composite theory exists.
- Manipulate terms.

My strategy: no-go theorems for distributive laws

Using composite theories:

- Choose two theories to compose.
- Assume composite theory exists.
- Manipulate terms.
- Derive contradiction of form $x = y$.

My strategy: no-go theorems for distributive laws

Using composite theories:

- Choose two theories to compose.
- Assume composite theory exists.
- Manipulate terms.
- Derive contradiction of form $x = y$.
- Conclusion: no such theory possible.

Image based on an image from Valter Bispo
Used under the creative commons licence

My strategy: no-go theorems for distributive laws

Using composite theories:

- Choose two theories to compose.
- Assume composite theory exists.
- Manipulate terms.
- Derive contradiction of form $x = y$.
- Conclusion: no such theory possible.
- List equations in the proof.

Image based on an image from Valter Bispo
Used under the creative commons licence

My strategy: no-go theorems for distributive laws

Using composite theories:

- Choose two theories to compose.
- Assume composite theory exists.
- Manipulate terms.
- Derive contradiction of form $x = y$.
- Conclusion: no such theory possible.
- List equations in the proof.
- $\Rightarrow$ No-go theorem.

Image based on an image from Valter Bispo
Used under the creative commons licence

The Boom Hierarchy

The Boom Hierarchy is a set of data structures:

Trees

↓

Lists

↓

Bags

↓

Sets

The Boom Hierarchy

The Boom Hierarchy is a set of data structures:

Trees $\downarrow$ Lists $\downarrow$ Bags $\downarrow$ Sets

Magmas: $\Sigma = \{0^{(0)}, +^{(2)}\}$, (unital equations)

The Boom Hierarchy

The Boom Hierarchy is a set of data structures:

Trees → Lists → Bags → Sets

Magmas: $\Sigma = \{0^{(0)}, +^{(2)}\}$, (unital equations)

↓ plus associativity

Monoids

The Boom Hierarchy

The Boom Hierarchy is a set of data structures:

Trees ↓ Lists ↓ Bags ↓ Sets

Magmas: $\Sigma = \{0^{(0)}, +^{(2)}\}$, (unital equations)

↓ plus associativity

Monoids

↓ plus commutativity

Commutative Monoids

The Boom Hierarchy

The Boom Hierarchy is a set of data structures:

Trees	Magmas:	$\Sigma = \{0^{(0)}, +^{(2)}\}$, (unital equations)
↓	↓ plus associativity	
Lists	Monoids	
↓	↓ plus commutativity	
Bags	Commutative Monoids	
↓	↓ plus idempotence	
Sets	Join-semilattices	

The Boom Hierarchy

The Boom Hierarchy is a set of data structures:

Trees	Magmas:
↓	↓ plus associativity
Lists	Monoids
↓	↓ plus commutativity
Bags	Commutative Monoids
↓	↓ plus idempotence
Sets	Join-semilattices

Why this hierarchy?

- Practical Monads
- Simple Theories
- Interesting Properties

A first look

The Boom Hierarchy

Possibility of compositions Column ∘ Row.

	Trees	Lists	Bags	Sets
Trees	N	N	Y	Y
Lists	N	N	Y	y
Bags	N	N	Y	Y
Sets	N	N	N	N

- Manes and Mulry 2007, 2008
- Klin and Salamanca 2018
- Zwart and Marsden 2019, 2020 (under review)

A first look

The Boom Hierarchy

Possibility of compositions Column ∘ Row.

	Trees	Lists	Bags	Sets
Trees	N	N	Y	Y
Lists	N	N	Y	y
Bags	N	N	Y	Y
Sets	N	N	N	N

The Non-Empty Boom Hierarchy

Possibility of compositions Column ∘ Row.

	Trees	Lists	Bags	Sets
Trees	Y	Y	Y	Y
Lists	Y	Y	Y	Y
Bags	Y	?	Y	Y
Sets	?	?	N	N

- Manes and Mulry 2007, 2008
- Klin and Salamanca 2018
- Zwart and Marsden 2019, 2020 (under review)

Extending the Boom hierarchy

- Boom Hierachy: 4 structures (8 if non-empty are considered)

Extending the Boom hierarchy

- Boom Hierachy: 4 structures (8 if non-empty are considered)
- Extension: all combinations of axioms gives 16 structures.

Extending the Boom hierarchy

- Boom Hierachy: 4 structures (8 if non-empty are considered)
- Extension: all combinations of axioms gives 16 structures.
  - Unit (U): Y/N

Extending the Boom hierarchy

- Boom Hierachy: 4 structures (8 if non-empty are considered)
- Extension: all combinations of axioms gives 16 structures.
  - Unit (U): Y/N
  - Associativity (A): Y/N

Extending the Boom hierarchy

- Boom Hierachy: 4 structures (8 if non-empty are considered)
- Extension: all combinations of axioms gives 16 structures.
  - Unit (U): Y/N
  - Associativity (A): Y/N
  - Commutativity (C): Y/N

Extending the Boom hierarchy

- Boom Hierachy: 4 structures (8 if non-empty are considered)
- Extension: all combinations of axioms gives 16 structures.
  - Unit (U): Y/N
  - Associativity (A): Y/N
  - Commutativity (C): Y/N
  - Idempotence (I): Y/N

Extending the Boom hierarchy

- Boom Hierachy: 4 structures (8 if non-empty are considered)
- Extension: all combinations of axioms gives 16 structures.
  - Unit (U): Y/N
  - Associativity (A): Y/N
  - Commutativity (C): Y/N
  - Idempotence (I): Y/N
- UAC stands for a structure with signature $\Sigma = \{0^{(0)}, +^{(2)}\}$ and the equations Unit, Associativity, Commutativity: bags.

A dramatic result

Combinations where both structures have units:

The Extended Boom Hierarchy (1/4)

Possibility of compositions Column ∘ Row.

	U	UI	UC	UCI	UA	UAI	UAC	UACI
U	N	N	N	N	N	N	Y	Y
UI	N	N	N	N	N	N	N	N
UC	N	N	N	N	N	N	Y	Y
UCI	N	N	N	N	N	N	N	N
UA	N	N	N	N	N	N	Y	Y
UAI	N	N	N	N	N	N	N	N
UAC	N	N	N	N	N	N	Y	Y
UACI	N	N	N	N	N	N	N	N

Venturing into unknown territory

The non-empty equivalents are more promising:

The Extended Boom Hierarchy (2/4)

Possibility of compositions Column ∘ Row.

	Ø	I	C	CI	A	AI	AC	ACI
Ø	Y		Y		Y		Y	Y
I		N	N	N		N	N	N
C	Y		Y				Y	Y
CI		N	N	N		N	N	N
A	Y				Y		Y	Y
AI		N	N	N		N	N	N
AC	Y						Y	Y
ACI		N	N	N		N	N	N

The full picture

Extended Boom Hierarchy, showing which compositions of form Column ∘ Row are possible.

Theories all consist of one binary operator, that is possibly idempotent (I), commutative (C), and/or associative (A), with possibly a constant that satisfies the unit equations. (U)
Y indicates a successfull composition, N indicates that the composition is impossible, empty cells represent unknowns. My own contributions have been highlighted in green.

	Ø	I	C	CI	A	AI	AC	ACI	U	UI	UC	UCI	UA	UAI	UAC	UACI
Ø	Y		Y		Y		Y	Y							Y	Y
I		N	N	N		N	N	N		N	N	N		N	N	N
C	Y		Y				Y	Y							Y	Y
CI		N	N	N		N	N	N		N	N	N		N	N	N
A	Y				Y		Y	Y							Y	Y
AI		N	N	N		N	N	N		N	N	N		N	N	N
AC	Y						Y	Y							Y	Y
ACI		N	N	N		N	N	N		N	N	N		N	N	N
U	Y						Y	Y	N	N	N	N	N	N	Y	Y
UI		N	N	N		N	N	N	N	N	N	N	N	N	N	N
UC	Y						Y	Y	N	N	N	N	N	N	Y	Y
UCI		N	N	N		N	N	N	N	N	N	N	N	N	N	N
UA	Y						Y	Y	N	N	N	N	N	N	Y	Y
UAI		N	N	N		N	N	N	N	N	N	N	N	N	N	N
UAC	Y						Y	Y	N	N	N	N	N	N	Y	Y
UACI		N	N	N		N	N	N	N	N	N	N	N	N	N	N

Lessons from the Boom hierarchy

- Idempotence/Units are bad.

Lessons from the Boom hierarchy

- Idempotence/Units are bad.
  (but not always)

Lessons from the Boom hierarchy

- Idempotence/Units are bad.

  (but not always)

  $x * x = x \qquad x * 1 = x$

Lessons from the Boom hierarchy

- Idempotence/Units are bad.
  (but not always)

$x * x = x \qquad x * 1 = x \qquad x \vee (x \wedge y) = x$

Lessons from the Boom hierarchy

- Idempotence/Units are bad.
  (but not always)

  $x * x = x \qquad x * 1 = x \qquad x \vee (x \wedge y) = x$

- Key property: reducing a term to a variable.

Lessons from the Boom hierarchy

- Idempotence/Units are bad.

  (but not always)

  $x * x = x \qquad x * 1 = x \qquad x \vee (x \wedge y) = x$

- Key property: reducing a term to a variable.
- **Conjecture**:

  Equations that reduce a term to a variable are necessary for distributive laws to fail.

  (but not sufficient)

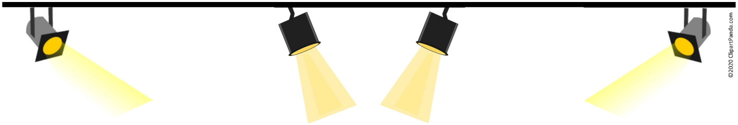
Predictions

Extended Boom Hierarchy, showing which compositions of form Column ◦ Row are possible.

Theories all consist of one binary operator, that is possibly idempotent (I), commutative (C), and/or associative (A), with possibly a constant that satisfies the unit equations. (U)
Y indicates a successfull composition, N indicates that the composition is impossible, empty cells represent unknowns. My own contributions have been highlighted in green.

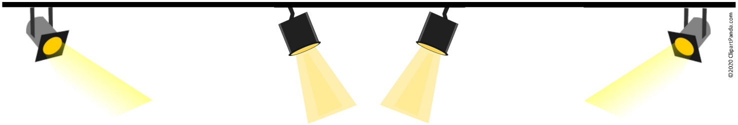
	Ø	I	C	CI	A	AI	AC	ACI	U	UI	UC	UCI	UA	UAI	UAC	UACI
Ø	Y		Y		Y		Y	Y							Y	Y
I		N	N	N		N	N	N		N	N	N		N	N	N
C	Y		Y				Y	Y							Y	Y
CI		N	N	N		N	N	N		N	N	N		N	N	N
A	Y				Y		Y	Y							Y	Y
AI		N	N	N		N	N	N		N	N	N		N	N	N
AC	Y						Y	Y							Y	Y
ACI		N	N	N		N	N	N		N	N	N		N	N	N
U	Y						Y	Y	N	N	N	N	N	N	Y	Y
UI		N	N	N		N	N	N	N	N	N	N	N	N	N	N
UC	Y						Y	Y	N	N	N	N	N	N	Y	Y
UCI		N	N	N		N	N	N	N	N	N	N	N	N	N	N
UA	Y						Y	Y	N	N	N	N	N	N	Y	Y
UAI		N	N	N		N	N	N	N	N	N	N	N	N	N	N
UAC	Y						Y	Y	N	N	N	N	N	N	Y	Y
UACI		N	N	N		N	N	N	N	N	N	N	N	N	N	N

When it's just too much



Too Many Constants Theorem

When it's just too much



Too Many Constants Theorem

But first, a proposition.

An important proposition

We need an interaction law:

Proposition (Multiplicative Zeroes)

Let $\mathbb{S}$ be an algebraic theory with a term s such that:

An important proposition

We need an interaction law:

Proposition (Multiplicative Zeroes)

Let $\mathbb{S}$ be an algebraic theory with a term s such that:

- s can be reduced to a variable via a substitution.

An important proposition

We need an interaction law:

Proposition (Multiplicative Zeroes)

Let $\mathbb{S}$ be an algebraic theory with a term s such that:

- s can be reduced to a variable via a substitution.

  e.g. $x * y$ with $*$ idempotent, $x \vee (y \wedge z)$ with absorption, etc.

An important proposition

We need an interaction law:

Proposition (Multiplicative Zeroes)

Let $\mathbb{S}$ be an algebraic theory with a term s such that:

- s can be reduced to a variable via a substitution.

  e.g. $x * y$ with $*$ idempotent, $x \vee (y \wedge z)$ with absorption, etc.

And let $\mathbb{T}$ be an algebraic theory with a constant 0 such that:

An important proposition

We need an interaction law:

Proposition (Multiplicative Zeroes)

Let $\mathbb{S}$ be an algebraic theory with a term s such that:

- s can be reduced to a variable via a substitution.
  e.g. $x * y$ with $*$ idempotent, $x \vee (y \wedge z)$ with absorption, etc.

And let $\mathbb{T}$ be an algebraic theory with a constant 0 such that:

$$t[f] =_{\mathbb{T}} 0 \Rightarrow t =_{\mathbb{T}} 0$$

An important proposition

We need an interaction law:

Proposition (Multiplicative Zeroes)

Let $\mathbb{S}$ be an algebraic theory with a term s such that:

- s can be reduced to a variable via a substitution.
  e.g. $x * y$ with $*$ idempotent, $x \vee (y \wedge z)$ with absorption, etc.

And let $\mathbb{T}$ be an algebraic theory with a constant 0 such that:

$$t[f] =_{\mathbb{T}} 0 \Rightarrow t =_{\mathbb{T}} 0$$

Any composite theory $\mathbb{U}$ of $\mathbb{T}$ after $\mathbb{S}$ has the following interaction:
For any $x \in \mathrm{var}(s)$:

$$s[0/x] =_{\mathbb{U}} 0.$$

Theorem (No-Go Theorem: Too Many Constants)

Let $\mathbb{S}$ be an algebraic theory with a term s such that:

- *s can be reduced to a variable via a substitution.*
- *s has **two** or more free variables.*

*And let $\mathbb{T}$ be an algebraic theory with **at least two constants** $0, 1$ such that for both constants:*

$$t[f] =_{\mathbb{T}} 0 \Rightarrow t =_{\mathbb{T}} 0 \qquad t[f] =_{\mathbb{T}} 1 \Rightarrow t =_{\mathbb{T}} 1$$

Then there exists no composite theory of $\mathbb{T}$ after $\mathbb{S}$.

Proof of the Too Many Constants Theorem

Proof.

Suppose that $\mathbb{U}$ is a composite theory of $\mathbb{T}$ after $\mathbb{S}$

Proof of the Too Many Constants Theorem

Proof.

Suppose that $\mathbb{U}$ is a composite theory of $\mathbb{T}$ after $\mathbb{S}$
Then by Proposition 1 we have:

DEPARTMENT OF COMPUTER SCIENCE UNIVERSITY OF OXFORD

Proof of the Too Many Constants Theorem

Proof.

Suppose that $\mathbb{U}$ is a composite theory of $\mathbb{T}$ after $\mathbb{S}$
Then by Proposition 1 we have:

$$s(x, y)$$

□

Proof of the Too Many Constants Theorem

Proof.

Suppose that $\mathbb{U}$ is a composite theory of $\mathbb{T}$ after $\mathbb{S}$
Then by Proposition 1 we have:

$$s(x, y)[0/x, 1/y]$$

□

Proof of the Too Many Constants Theorem

Proof.

Suppose that $\mathbb{U}$ is a composite theory of $\mathbb{T}$ after $\mathbb{S}$
Then by Proposition 1 we have:

$$0 =_{\mathbb{U}} s(x, y)[0/x, 1/y] =_{\mathbb{U}} 1.$$

□

Proof of the Too Many Constants Theorem

Proof.

Suppose that $\mathbb{U}$ is a composite theory of $\mathbb{T}$ after $\mathbb{S}$
Then by Proposition 1 we have:

$$0 =_{\mathbb{U}} s(x, y)[0/x, 1/y] =_{\mathbb{U}} 1.$$

Contradiction. So $\mathbb{U}$ cannot be a composite of $\mathbb{T}$ after $\mathbb{S}$. □

Another extension of the Boom hierarchy

Iterating Compositions in the Boom Hierarchy

Possibility of compositions Column ∘ Row.

	Trees	Lists	Bags	Sets	BT	BL	BB	ST	SL	SB
Trees	N	N	Y	Y						
Lists	N	N	Y	Y						
Bags	N	N	Y	Y						
Sets	N	N	N	N						
BT										
BL										
BB										
ST										
SL										
SB										

Another extension of the Boom hierarchy

Iterating Compositions in the Boom Hierarchy

Possibility of compositions Column ∘ Row.

	Trees	Lists	Bags	Sets	BT	BL	BB	ST	SL	SB
Trees	N	N	Y	Y	N	N	N	N	N	N
Lists	N	N	Y	Y	N	N	N	N	N	N
Bags	N	N	Y	Y	N	N	N	N	N	N
Sets	N	N	N	N	N	N	N	N	N	N
BT					N	N	N	N	N	N
BL					N	N	N	N	N	N
BB					N	N	N	N	N	N
ST					N	N	N	N	N	N
SL					N	N	N	N	N	N
SB					N	N	N	N	N	N

Another extension of the Boom hierarchy

Iterating Compositions in the Boom Hierarchy

Possibility of compositions Column ∘ Row.

	Trees	Lists	Bags	Sets	BT	BL	BB	ST	SL	SB
Trees	N	N	Y	Y	N	N	N	N	N	N
Lists	N	N	Y	Y	N	N	N	N	N	N
Bags	N	N	Y	Y	N	N	N	N	N	N
Sets	N	N	N	N	N	N	N	N	N	N
BT	N	N	N	N	N	N	N	N	N	N
BL	N	N	N	N	N	N	N	N	N	N
BB	N	N	N	N	N	N	N	N	N	N
ST	N	N	N	N	N	N	N	N	N	N
SL	N	N	N	N	N	N	N	N	N	N
SB	N	N	N	N	N	N	N	N	N	N

Conclusion

- Not all monads compose via a distributive law.

- https://www.cs.ox.ac.uk/people/maaike.zwart/
- maaike.annebeth@gmail.com

Conclusion

- Not all monads compose via a distributive law.
- Boom hierarchy provides some intuition.

- https://www.cs.ox.ac.uk/people/maaike.zwart/
- maaike.annebeth@gmail.com

Conclusion

- Not all monads compose via a distributive law.
- Boom hierarchy provides some intuition.
- Reducing a term to a variable key property for no-go theorems.

- `https://www.cs.ox.ac.uk/people/maaike.zwart/`
- maaike.annebeth@gmail.com

Conclusion

- Not all monads compose via a distributive law.
- Boom hierarchy provides some intuition.
- Reducing a term to a variable key property for no-go theorems.
- Too many constants / multiplicative zeroes prevent iterated distributive laws within the Boom hierarchy.

- `https://www.cs.ox.ac.uk/people/maaike.zwart/`
- maaike.annebeth@gmail.com