The Theory of Call-by-Value Solvability

Beniamino Accattoli[1] **Giulio Guerrieri**[2]

[1]Inria Saclay, France

[2]Huawei Edinburgh Research Centre, United Kingdom

June 20, 2022

Outline

Table of Contents

Meaningful and meaningless in the λ-calculus

A semantics of the (untyped) λ-calculus $\approx$ an equational theory over λ-terms.

Which equational theories collapse all meaningless λ-terms?
What does it mean being meaningless?

Meaningful and meaningless in the λ-calculus

A semantics of the (untyped) λ-calculus $\approx$ an equational theory over λ-terms.

- It induces some equivalence classes on λ-terms.
- λ-terms in the same equivalence class share the same "meaning".

Which equational theories collapse all meaningless λ-terms?
What does it mean being meaningless?

Meaningful and meaningless in the λ-calculus

A semantics of the (untyped) λ-calculus $\approx$ an equational theory over λ-terms.

- It induces some equivalence classes on λ-terms.
- λ-terms in the same equivalence class share the same "meaning".

A reasonable approach to give a meaning to λ-terms:

Which equational theories collapse all meaningless λ-terms?
What does it mean being meaningless?

Meaningful and meaningless in the λ-calculus

A semantics of the (untyped) λ-calculus $\approx$ an equational theory over λ-terms.

- It induces some equivalence classes on λ-terms.
- λ-terms in the same equivalence class share the same "meaning".

A reasonable approach to give a meaning to λ-terms:

- Each equivalence class must be stable by β-conversion;
- There are many different equivalence classes of meaningful λ-terms;
- Collapse: all meaningless λ-terms should be equated.

Which equational theories collapse all meaningless λ-terms?
What does it mean being meaningless?

A naive theory: meaningless = non-normalizable

Idea:

1. A β-normal form is the result of a computation;
2. β-normalizing λ-terms are meaningful ($\approx$ defined partial recursive functions);
3. β-diverging λ-terms are meaningless ($\approx$ undefined partial recursive functions).

A naive theory: meaningless = non-normalizable

Idea:

1. A β-normal form is the result of a computation;
2. β-normalizing λ-terms are meaningful ($\approx$ defined partial recursive functions);
3. β-diverging λ-terms are meaningless ($\approx$ undefined partial recursive functions).

Drawbacks of collapsing all β-diverging λ-terms [Bar74,Wad76]:

1. The representation of partial recursive functions is not stable by composition;
2. Inconsistency: the theory equates all λ-terms! (it collapses everything!)

A naive theory: meaningless = non-normalizable

Idea:

1. A β-normal form is the result of a computation;
2. β-normalizing λ-terms are meaningful ($\approx$ defined partial recursive functions);
3. β-diverging λ-terms are meaningless ($\approx$ undefined partial recursive functions).

Drawbacks of collapsing all β-diverging λ-terms [Bar74,Wad76]:

1. The representation of partial recursive functions is not stable by composition;
2. Inconsistency: the theory equates all λ-terms! (it collapses everything!)

Moral:

1. Being β-normalizable is not a meaningful predicate.
2. β-normalizing terms are not the only meaningful λ-terms.

A sensible theory: meaningless = unsolvable

Definition: A λ-term t is solvable if there is a head context H sending $H\langle t\rangle$ to the identity $I = \lambda x.x$, that is, such that $H\langle t\rangle \to_\beta^* I$.

Idea: A solvable term t might be divergent but all its diverging sub-terms are removable without discarding the whole t.

A sensible theory: meaningless = unsolvable

Definition: A λ-term t is solvable if there is a head context H sending $H\langle t\rangle$ to the identity $I = \lambda x.x$, that is, such that $H\langle t\rangle \to_\beta^* I$.

Idea: A solvable term t might be divergent but all its diverging sub-terms are removable without discarding the whole t.

Example: Let $\delta = \lambda z.zz$. Then, $\Omega = \delta\delta$ is unsolvable. But $x\Omega$ is solvable!
Let $H = (\lambda x.\langle\cdot\rangle)\lambda y.I$: $H\langle x\Omega\rangle = (\lambda x.x\Omega)\lambda y.I \to_\beta (\lambda y.I)\Omega \to_\beta I$.

A sensible theory: meaningless = unsolvable

Definition: A λ-term t is solvable if there is a head context H sending $H\langle t\rangle$ to the identity $I = \lambda x.x$, that is, such that $H\langle t\rangle \to_\beta^* I$.

Idea: A solvable term t might be divergent but all its diverging sub-terms are removable without discarding the whole t.

Example: Let $\delta = \lambda z.zz$. Then, $\Omega = \delta\delta$ is unsolvable. But $x\Omega$ is solvable!
Let $H = (\lambda x.\langle\cdot\rangle)\lambda y.I$: $H\langle x\Omega\rangle = (\lambda x.x\Omega)\lambda y.I \to_\beta (\lambda y.I)\Omega \to_\beta I$.

Theorem [Bar74]: Collapsing all unsolvable terms is consistent (sensible theories).

Examples: $\mathcal{H}$, theories induced by models (Scott's D_∞, relational semantics, etc.).

Characterizations of solvability: a beautiful theory

Definition of solvability is not handy (how to find an head context?)

Characterizations of solvability: a beautiful theory

Definition of solvability is not handy (how to find an head context?)

Characterizations of solvability: a beautiful theory

Definition of solvability is not handy (how to find an head context?)

Theorem [Operational characterization, Bar74]: t is solvable iff head reduction terminates on t.

Corollary: The class of solvable terms strictly includes the β-normalizing ones. Morally, unsolvable means "heavily divergent".

Characterizations of solvability: a beautiful theory

Definition of solvability is not handy (how to find an head context?)

Theorem [Operational characterization, Bar74]: t is solvable iff head reduction terminates on t.

Corollary: The class of solvable terms strictly includes the β-normalizing ones. Morally, unsolvable means "heavily divergent".

Theorem [Type-theoretic characterizations, [CopDez80,deC07]: t is solvable iff t is typable in a (idempotent or non-idemptotent) intersection type system.

Characterizations of solvability: a beautiful theory

Definition of solvability is not handy (how to find an head context?)

Theorem [Operational characterization, Bar74]: t is solvable iff head reduction terminates on t.

Corollary: The class of solvable terms strictly includes the β-normalizing ones. Morally, unsolvable means "heavily divergent".

Theorem [Type-theoretic characterizations, [CopDez80,deC07]: t is solvable iff t is typable in a (idempotent or non-idemptotent) intersection type system.

Theorem [Genericity, Bar84]: Let t be unsolvable, u be β-normal, C be a context. If $C\langle t\rangle \to_\beta^* u$ then $C\langle s\rangle \to_\beta^* u$ for every term s.

Idea: $C\langle t\rangle$ normalizes and has a unsolvable subterm t, so t is discarded.

Table of Contents

Plotkin's Call-by-Value λ-calculus [Plo75]

Terms $s, t, u ::= v \mid tu$ Values $v ::= x \mid \lambda x.t$

CbV reduction $(\lambda x.t)v \to_{\beta_v} t\{v/x\}$

It is closer to real implementation of most programming languages.
The semantics is completely different from standard (Call-by-Name) λ-calculus.

Plotkin's Call-by-Value λ-calculus [Plo75]

Terms $s, t, u ::= v \mid tu$ Values $v ::= x \mid \lambda x.t$

CbV reduction $(\lambda x.t)v \rightarrow_{\beta_v} t\{v/x\}$

It is closer to real implementation of most programming languages.
The semantics is completely different from standard (Call-by-Name) λ-calculus.

Plotkin's Call-by-Value λ-calculus [Plo75]

Terms $s, t, u ::= v \mid tu$ Values $v ::= x \mid \lambda x.t$

CbV reduction $(\lambda x.t)v \to_{\beta_v} t\{v/x\}$

It is closer to real implementation of most programming languages.
The semantics is completely different from standard (Call-by-Name) λ-calculus.

Examples:

1. $(\lambda x.\delta)(xx)\delta$ is β_v-normal but β-divergent!

Call-by-Value solvability

Definition: A head context is a context defined by $H ::= \langle\cdot\rangle \mid \lambda x.H \mid Ht$.

A λ-term t is β_v-solvable if there is a head context H sending $H\langle t\rangle$ to the identity $I = \lambda x.x$, that is, such that $H\langle t\rangle \to_{\beta_v}{}^{*} I$.

Call-by-Value solvability

Definition: A head context is a context defined by $H ::= \langle\cdot\rangle \mid \lambda x.H \mid Ht$.

A λ-term t is β_v-solvable if there is a head context H sending $H\langle t\rangle$ to the identity $I = \lambda x.x$, that is, such that $H\langle t\rangle \to_{\beta_v}{}^{*} I$.

Examples:

1. $x\Omega$ is β_v-unsolvable, because Ω cannot be erased (but it is β-solvable).
2. $(\lambda x.\delta)(xx)\delta$ is β_v-normal but β_v-unsolvable.
3. No operational characterization of β_v-solvability inside Plotkin's calculus!

Call-by-Value solvability

Definition: A head context is a context defined by $H ::= \langle\cdot\rangle \mid \lambda x.H \mid Ht$.

A λ-term t is β_v-solvable if there is a head context H sending $H\langle t\rangle$ to the identity $I = \lambda x.x$, that is, such that $H\langle t\rangle \to_{\beta_v}{}^* I$.

Examples:

1. $x\Omega$ is β_v-unsolvable, because Ω cannot be erased (but it is β-solvable).
2. $(\lambda x.\delta)(xx)\delta$ is β_v-normal but β_v-unsolvable.
3. No operational characterization of β_v-solvability inside Plotkin's calculus!

What a mess!

Alternative CbV λ-calculus: Value Substitution [AccPao12]

Terms: $s, t, u ::= v \mid tu \mid t[u/x]$ Values: $v ::= x \mid \lambda x.t$

Substitution contexts: $L ::= [t_1/x_1] \ldots [t_n/x_n]$

Reductions: $(\lambda x.t)Lu \to_m t[u/x]L$ $t[vL/x] \to_e t\{v/x\}L$

Alternative CbV λ-calculus: Value Substitution [AccPao12]

Terms: $s, t, u ::= v \mid tu \mid t[u/x]$ Values: $v ::= x \mid \lambda x.t$

Substitution contexts: $L ::= [t_1/x_1] \ldots [t_n/x_n]$

Reductions: $(\lambda x.t)Lu \to_m t[u/x]L$ $t[vL/x] \to_e t\{v/x\}L$

1. β_v-reduction can be simulated into VSC.

$$(\lambda x.t)v \to_m t[v/x] \to_e t\{v/x\}$$

2. VSC extends β_v-reduction:

$$(\lambda x.\delta)(xx)\delta \to_m \delta[xx/x]\delta \to_m (zz)[\delta/z][xx/x] \to_e \delta\delta[xx/x] \to \cdots$$

Operational internal characterization of VSC-Solvability

Theorem [AccPao12]: t is VSC-solvable iff solving reduction terminates on t.

Solving reduction: restriction of VSC not firing under λ on the left of application.

Corollary: The set of VSC-scrutable terms strictly includes the VSC-solvable ones.

Operational internal characterization of VSC-Solvability

Theorem [AccPao12]: t is VSC-solvable iff solving reduction terminates on t.

Solving reduction: restriction of VSC not firing under λ on the left of application.

Theorem [AccPao12]: t is VSC-scrutable iff weak reduction terminates on t.

Weak reduction: restriction of VSC not firing under λ.
Scrutability: t is VSC-scrutable (aka VSC-potentially valuable) if there are values $v, v_1, \ldots, v_n$ such that $t\{v_1/x_1, \ldots, v_n/x_n\} \rightarrow^*_{VSC} v$.

Corollary: The set of VSC-scrutable terms strictly includes the VSC-solvable ones.

Table of Contents

Some results

Theorem [Robustness]:

1. t is VSC-scrutable iff t is β_v-scrutable.
2. t is VSC-solvable iff t is β_v-solvable.

The notions are robust in CbV, do not depend on the (CbV) calculus.
VSC is a tool to study them!

Some results

Theorem [Robustness]:

1. t is VSC-scrutable iff t is β_v-scrutable.
2. t is VSC-solvable iff t is β_v-solvable.

The notions are robust in CbV, do not depend on the (CbV) calculus.
VSC is a tool to study them!

Theorem [(Un-)Collapsibility]:

1. Collapsing all CbV-unsolvable terms is inconsistent.
2. Collapsing all CbV-inscrutable terms is consistent.

In CbV, meaningless = unscrutable. In CbN, meaningless = unsolvable.

Some results

Theorem [Robustness]:

1. t is VSC-scrutable iff t is β_v-scrutable.
2. t is VSC-solvable iff t is β_v-solvable.

The notions are robust in CbV, do not depend on the (CbV) calculus.
VSC is a tool to study them!

Theorem [(Un-)Collapsibility]:

1. Collapsing all CbV-unsolvable terms is inconsistent.
2. Collapsing all CbV-inscrutable terms is consistent.

In CbV, meaningless = unscrutable. In CbN, meaningless = unsolvable.

Theorem [Non Genericity]: Genericity does not hold with CbV solvability.
Conjecture [Genericity]: Genericity does hold with CbV scrutability.

Type-theoretic characterization of solvability/scrutability

Theorem

1. t is CbV-scrutable iff t is typable in a (suitable) non-idempotent intersection type system.
2. t is CbV-solvable iff t is typable in a (suitable) restriction of the non-idempotent intersection type system.