Constructive Galois Connections

David Darais
University of Maryland

David Van Horn
University of Maryland
You

prog. λ
Static Analysis

You

prog. λ

[Diagram showing a process where a person interacts with a program λ and sends it to a static analysis process]
You

Static Analysis

Correctness Proof
Static Analysis

Correctness Proof
Static Analysis

Correctness Proof
Abstract Interpretation

Static Analysis

Correctness Proof
Abstract Interpretation

Spec

Static Analysis

Correctness Proof
Abstract Interpretation

Spec
Static Analysis
Correctness Proof
Abstract Interpretation

Calcutational Design

Spec

Static Analysis

Correctness Proof
The Dream

Abstract Interpreters

\{ Synthesized specification
    Correct by construction
    Certified Implementation \}
The Calculational Design of a Generic Abstract Interpreter

Patrick COUSOT
LIENS, Département de Mathématiques et Informatique
École Normale Supérieure, 45 rue d’Ulm, 75230 Paris cedex 05, France

Abstract. We present in extenso the calculation-based development of a generic compositional reachability static analyzer for a simple imperative programming language by abstract interpretation of its formal rule-based/structured small-step operational semantics.

Contents

1. Introduction 3
2. Definitions 4
3. Values 5
   3.1 Machine integers ................................................. 5
   3.2 Errors ................................................................. 5
4. Properties of Values 5
5. Abstract Properties of Values 6
   5.1 Galois connection based abstraction ............................. 6
   5.2 Componentwise abstraction of sets of pairs ..................... 7
   5.3 Initialization and simple sign abstraction ....................... 7
   5.4 Initialization and interval abstraction .......................... 8
   5.5 Algebra of abstract properties of values ....................... 9
6. Environments 10
   6.1 Concrete environments ............................................ 10
   6.2 Properties of concrete environments ........................... 10
   6.3 Nonrelational abstraction of environment properties .......... 10
   6.4 Algebra of abstract environments ................................ 12
7. Semantics of Arithmetic Expressions 13
   7.1 Abstract syntax of arithmetic expressions ..................... 13
   7.2 Machine arithmetics ................................................. 13
   7.3 Operational semantics of arithmetic expressions ............... 13
   7.4 Forward collecting semantics of arithmetic expressions ........ 14
   7.5 Backward collecting semantics of arithmetic expressions ...... 14
8. Abstract Interpretation of Arithmetic Expressions 15
   8.1 Lifting Galois connections at higher-order .................... 15
The emphasis in these notes [has been the] correctness of the design by calculus. The mechanized verification [of this technique] can be foreseen with automatic extraction of a correct program from its correctness proof.

—Patrick Cousot [Monograph 1999]
THÈSE

présentée

devant l'Université de Rennes 1

pour obtenir

le grade de : DOCTEUR DE L'UNIVERSITÉ DE RENNES 1
Mention INFORMATIQUE

par

David Pichardie

Équipe d'accueil : Londe (InriaRennes)
École Doctorale : Matise
Composante universitaire : IFSIC

Titre de la thèse :

Interprétation abstraite en logique intuitionniste :
extraction d'analyseurs Java certifiés

Soutenue le 6 décembre 2006 devant la commission d'examen

M. Jean-Pierre Botître - Président
M. Patrick Consot - Rapporteurs
M. Xavier Lévy
Mme. Christine Baulin-Mohring - Examinateurs
M. David Schmidt
M. Thomas Jensen - Directeurs
M. David Charla
Our framework loses an important property of the standard framework: the ability to derive a correct approximation from its specification.

... It seems interesting to find a framework for deriving approximations, while remaining easily formalizable in Coq.

if (b) {x = 10} else {x = 20}
if (b) \{x = 10\} else \{x = 20\}

x ∈ \{10, 20\}
\[ \begin{align*} 
\text{if (b) } & \{x = 10\} \ \text{else } \{x = 20\} \\
\end{align*} \]
\[ \text{if } (b) \{ x = 10 \} \text{ else } \{ x = 20 \} \]

\[ x \in \{ 10, 20 \} \]

\[ x \in \{ 10, \ldots, 20 \} \]

\[ x \in \langle 10, 20 \rangle \]
... if (b) {x = 10} else {x = 20} ...

\[ \mathcal{O}(\mathbb{Z}) \]

\[ \mathbb{Z} \times \mathbb{Z} \]

\[ x \in \{10, 20\} \]

\[ x \in \{10, \ldots, 20\} \]

\[ x \in \langle 10, 20 \rangle \]
\[ \text{if } (b) \{ x = 10 \} \text{ else } \{ x = 20 \} \]

\[ \wp(\mathbb{Z}) \]

\[ x \in \{ 10, 20 \} \]

\[ x \in \{ 10, \ldots, 20 \} \]

\[ x \in \langle 10, 20 \rangle \]

Undecidable

Decidable
Classical Reasoning \( \varphi(\mathbb{Z}) \)

Program Extraction \( \mathbb{Z} \times \mathbb{Z} \)
\[
\alpha(\text{eval}[n])(\rho^#) \\
\text{defn of } \alpha \Downarrow \\
= \alpha^I(\text{eval}[n](\gamma^R(\rho^#))) \\
\text{defn of } \text{eval}[n] \Downarrow \\
= \alpha^I(\{i \mid \rho \vdash n \rightarrow i\}) \\
\text{defn of } \vdash \rightarrow \Downarrow \\
= \alpha^I(\{i\}) \\
\text{defn of } \text{eval}^#[n] \Downarrow \\
\Delta \equiv \text{eval}^#[n](\rho^#)
\]
\[ \alpha(\text{eval}[n])(\rho\#) \]

\[ \vdash \text{defn of } \alpha \vdash \]
\[ = \alpha^I(\text{eval}[n](\gamma^R(\rho\#))) \]

\[ \vdash \text{defn of } \text{eval}[n] \vdash \]
\[ = \alpha^I(\{i \mid \rho \vdash n \mapsto i\}) \]

\[ \vdash \text{defn of } _\vdash \vdash \vdash \]
\[ = \alpha^I(\{i\}) \]

\[ \vdash \text{defn of } \text{eval}^\#[n] \vdash \]
\[ \triangleq \text{eval}^\#[n](\rho\#) \]
Four Stories

Direct Verification

Abstract Interpretation

✗ calculate

✗ mechanize
Four Stories

Direct Verification

Abstract Interpretation

Kleisli GCs

Constructive GCs

✗ calculate

✗ mechanize

✓ calculate

½ mechanize

✓ calculate

✓ mechanize
Direct Verification
Direct Verification

\[\text{succ} : \mathbb{N} \rightarrow \mathbb{N}\]
Direct Verification

\[ \text{succ} : \mathbb{N} \rightarrow \mathbb{N} \quad \mathbb{P} = \{E, 0\} \]
Direct Verification

\[ \text{succ} : \mathbb{N} \to \mathbb{N} \quad \mathbb{P} = \{E, 0\} \]

\[ \text{flip} : \mathbb{P} \to \mathbb{P} \]
\[ \text{flip}(E) = 0 \]
\[ \text{flip}(0) = E \]
Direct Verification

\[
\begin{align*}
succ & : \mathbb{N} \to \mathbb{N} \\
\mathbb{P} & = \{E, O\}
\end{align*}
\]

\[
\begin{align*}
\text{flip} & : \mathbb{P} \to \mathbb{P} \\
\text{flip}(E) & = 0 \\
\text{flip}(O) & = E \\
\end{align*}
\]

\[
\begin{align*}
\left\lceil \_ \right\rceil & : \mathbb{P} \to \wp(\mathbb{N}) \\
\left\lceil E \right\rceil & = \{ n \mid \text{even}(n) \} \\
\left\lceil O \right\rceil & = \{ n \mid \text{odd}(n) \}
\end{align*}
\]
Direct Verification

\[\text{succ} : \mathbb{N} \rightarrow \mathbb{N} \quad \text{P} = \{E, O\}\]

\[\text{flip} : \text{P} \rightarrow \text{P} \quad \boxed{[\_]} : \text{P} \rightarrow \wp(\mathbb{N})\]
\[\text{flip}(E) = 0 \quad \boxed{[E]} = \{ n \mid \text{even}(n) \} \]
\[\text{flip}(O) = E \quad \boxed{[O]} = \{ n \mid \text{odd}(n) \} \]

\[\text{sound} : n \in [p] \implies \text{succ}(n) \in [\text{flip}(p)]\]
Direct Verification

✓ flip can be extracted and executed
✓ \[ \_ \] can be mechanized effectively

✗ Is flip optimal?
✗ How to derive flip from succ?
Four Stories

Direct Verification

Abstract Interpretation

Kleisli GCs

Constructive GCs

✗ calculate

✗ mechanize

✓ calculate

½ mechanize

✓ calculate

✓ mechanize
Abstract Interpretation
Abstract Interpretation
Abstract Interpretation

\[ \wp(\mathbb{N}) \]

\{1\}

\{1,3\}

odd(n)
Abstract Interpretation

\[ \mathcal{P}(\mathbb{N}) \]

- \{1\}
- \{1, 3\}
- odd(n)

\[ \mathcal{P}(\mathbb{P}) \]

- \{\}\n- \{0\}
- \{E, 0\}
Abstract Interpretation

\[ \wp(\mathbb{N}) \]

\[ \wp(\mathbb{P}) \]

\{1\} \quad \{\} \quad \alpha

\{1,3\} \quad \alpha

\text{odd}(n) \quad \{0\}

\{\} \quad \{E,0\}
Abstract Interpretation

\[ \wp(\mathbb{N}) \]

\[ \{1\} \]

\[ \{1,3\} \]

\[ \text{odd}(n) \]

\[ \wp(\mathbb{P}) \]

\[ \{\} \]

\[ \{0\} \]

\[ \{E,0\} \]
Abstract Interpretation

\[ \wp(\mathbb{P}) \]

\[ \{ O \} \]

\[ \{ E, O \} \]

\[ \{ 1 \} \]

\[ \{ 1, 3 \} \]

\[ \text{odd}(n) \]

\[ \wp(\mathbb{N}) \]

\[ \mathbb{N} \subseteq \gamma(\alpha(\mathbb{N})) \]
Abstract Interpretation

\[ \wp(\mathbb{P}) \]

\{1\}
\{1,3\}
\text{odd}(n)

\{\} \subseteq \wp(\mathbb{P})
\{0\} \subseteq \wp(\mathbb{P})
\{E,0\} \subseteq \wp(\mathbb{P})

\mathbb{N} \subseteq \gamma(\alpha(\mathbb{N}))
\alpha(\gamma(\mathbb{P})) \subseteq \mathbb{P}
Abstract Interpretation

\[ N \subseteq \gamma(\alpha(N)) \land \alpha(\gamma(P)) \subseteq P \]

\[ N \subseteq \gamma(P) \iff \alpha(N) \subseteq P \]
Abstract Interpretation

N ∈ $\mathcal{A}(N)$
P ∈ $\mathcal{A}(P)$

"P is sound for N"

$\alpha(N) \subseteq P$
Abstract Interpretation

\[ f^N \in \wp(\mathbb{N}) \to \wp(\mathbb{N}) \]
\[ f^P \in \wp(\mathbb{P}) \to \wp(\mathbb{P}) \]

"\( f^P \) is sound for \( f^N \)"

\[ \alpha \circ f^N \circ \gamma \subseteq f^P \]
Abstract Interpretation

\[\alpha : \mathcal{P}(\mathbb{N}) \rightarrow \mathcal{P}(\mathbb{P})\]

\[\alpha(\mathbb{N}) := \{\text{parity}(n) \mid n \in \mathbb{N}\}\]
Abstract Interpretation

\[ \alpha : \wp(\mathbb{N}) \to \wp(\mathbb{P}) \]
\[ \alpha(N) := \{ \text{parity}(n) \mid n \in \mathbb{N} \} \]

\[ \gamma : \wp(\mathbb{P}) \to \wp(\mathbb{N}) \]
\[ \gamma(P) := \{ n \mid p \in P \land n \in \llbracket p \rrbracket \} \]
Abstract Interpretation

\[ \alpha : \wp(\mathbb{N}) \to \wp(\mathbb{P}) \]
\[ \alpha(\mathbb{N}) = \{ \text{parity}(n) \mid n \in \mathbb{N} \} \]
\[ \alpha \approx \text{map} (\text{parity}) \]

\[ \gamma : \wp(\mathbb{P}) \to \wp(\mathbb{N}) \]
\[ \gamma(\mathbb{P}) = \{ n \mid p \in \mathbb{P} \land n \in \llbracket p \rrbracket \} \]
\[ \gamma \approx \text{extend} (\llbracket \_ \rrbracket) \]
Abstract Interpretation

\[ \alpha : \mathcal{P}(\mathbb{N}) \to \mathcal{P}(\mathbb{P}) \]

\[ \gamma : \mathcal{P}(\mathbb{P}) \to \mathcal{P}(\mathbb{N}) \]

\[ \text{succ} : \mathbb{N} \to \mathbb{N} \]

\[ \text{flip} : \mathbb{P} \to \mathbb{P} \]
Abstract Interpretation

\[ \alpha : \wp(\mathbb{N}) \rightarrow \wp(\mathbb{P}) \]
\[ \gamma : \wp(\mathbb{P}) \rightarrow \wp(\mathbb{N}) \]
\[ \text{succ} : \mathbb{N} \rightarrow \mathbb{N} \]
\[ \text{flip} : \mathbb{P} \rightarrow \mathbb{P} \]

\[ \text{↑succ} : \wp(\mathbb{N}) \rightarrow \wp(\mathbb{N}) \]
\[ \text{↑succ}(\mathbb{N}) = \{\text{succ}(n) \mid n \in \mathbb{N}\} \]

\[ \text{↑flip} : \wp(\mathbb{P}) \rightarrow \wp(\mathbb{P}) \]
\[ \text{↑flip}(\mathbb{P}) = \{\text{flip}(p) \mid p \in \mathbb{P}\} \]
Abstract Interpretation

\[ \alpha : \wp(\mathbb{N}) \rightarrow \wp(\mathbb{P}) \quad \text{succ : } \mathbb{N} \rightarrow \mathbb{N} \]

\[ \gamma : \wp(\mathbb{P}) \rightarrow \wp(\mathbb{N}) \quad \text{flip : } \mathbb{P} \rightarrow \mathbb{P} \]

\[ \uparrow \text{succ} : \wp(\mathbb{N}) \rightarrow \wp(\mathbb{N}) \]

\[ \uparrow \text{succ}(\mathbb{N}) = \{ \text{succ}(n) \mid n \in \mathbb{N} \} \]

\[ \uparrow \text{flip} : \wp(\mathbb{P}) \rightarrow \wp(\mathbb{P}) \]

\[ \uparrow \text{flip}(\mathbb{P}) = \{ \text{flip}(p) \mid p \in \mathbb{P} \} \]

sound : \[ \alpha(\uparrow \text{succ}(\gamma(\mathbb{P}))) \subseteq \uparrow \text{flip}(\mathbb{P}) \]
Abstract Interpretation

\[ \alpha : \mathcal{P}(\mathbb{N}) \rightarrow \mathcal{P}(\mathcal{P}) \]
\[ \gamma : \mathcal{P}(\mathcal{P}) \rightarrow \mathcal{P}(\mathbb{N}) \]
\[ \text{succ} : \mathbb{N} \rightarrow \mathbb{N} \]
\[ \text{flip} : \mathcal{P} \rightarrow \mathcal{P} \]

\[ \uparrow \text{succ} : \mathcal{P}(\mathbb{N}) \rightarrow \mathcal{P}(\mathbb{N}) \]
\[ \uparrow \text{succ}(\mathbb{N}) = \{\text{succ}(n) \mid n \in \mathbb{N}\} \]

\[ \uparrow \text{flip} : \mathcal{P}(\mathcal{P}) \rightarrow \mathcal{P}(\mathcal{P}) \]
\[ \uparrow \text{flip}(\mathcal{P}) = \{\text{flip}(p) \mid p \in \mathcal{P}\} \]

<table>
<thead>
<tr>
<th>sound</th>
<th>( \alpha(\uparrow \text{succ}(\gamma(\mathcal{P}))) \subseteq \uparrow \text{flip}(\mathcal{P}) )</th>
</tr>
</thead>
<tbody>
<tr>
<td>optimal</td>
<td>( \alpha(\uparrow \text{succ}(\gamma(\mathcal{P}))) = \uparrow \text{flip}(\mathcal{P}) )</td>
</tr>
</tbody>
</table>
Abstract Interpretation

optimal : $\alpha(\uparrow\text{succ}(\gamma(P))) = \uparrow\text{flip}(P)$
Abstract Interpretation

calc : α(↑succ(γ(P)))) = ... ≅ ↑flip(P)
Abstract Interpretation

calc : $\alpha(\uparrow \text{succ}(\gamma(P))) = \ldots \triangleq \uparrow \text{flip}(P)$

$\alpha(\uparrow \text{succ}(\gamma(\{E\})))$
Abstract Interpretation

calc : \( \alpha(\uparrow\text{succ}(\gamma(P))) = \ldots \triangleq \uparrow\text{flip}(P) \)

\[
\begin{align*}
\alpha(\uparrow\text{succ}(\gamma(\{E\}))) \\
= \alpha(\uparrow\text{succ}(\{n \mid \text{even}(n)\}))
\end{align*}
\]
Abstract Interpretation

calc : \( \alpha(\uparrow\text{succ}(\gamma(P))) = \ldots \upl \uparrow\text{flip}(P) \)

\[
\begin{align*}
\alpha(\uparrow\text{succ}(\gamma(\{E\}))) &= \alpha(\uparrow\text{succ}(\{n \mid \text{even}(n)\})) \\
&= \alpha(\{\text{succ}(n) \mid \text{even}(n)\})
\end{align*}
\]
Abstract Interpretation

calc : \( \alpha(\uparrow\text{succ}(\gamma(P))) = \ldots \upmodels \uparrow\text{flip}(P) \)

\[
\begin{align*}
\alpha(\uparrow\text{succ}(\gamma(\{E\}))) &= \alpha(\uparrow\text{succ}(\{n \mid \text{even}(n)\})) \\
&= \alpha(\{\text{succ}(n) \mid \text{even}(n)\}) \\
&= \alpha(\{n \mid \text{odd}(n)\})
\end{align*}
\]
Abstract Interpretation

calc : $\alpha(\uparrow\text{succ}(\gamma(P))) = \ldots \triangleq \uparrow\text{flip}(P)$

$$
\begin{align*}
\alpha(\uparrow\text{succ}(\gamma(\{E\})))
&= \alpha(\uparrow\text{succ}(\{n \mid \text{even}(n)\})) \\
&= \alpha(\{\text{succ}(n) \mid \text{even}(n)\}) \\
&= \alpha(\{n \mid \text{odd}(n)\}) \\
&= \{0\}
\end{align*}
$$
Abstract Interpretation

\[ \text{calc : } \alpha(\uparrow\text{succ}(\gamma(P))) = \ldots \triangleq \uparrow\text{flip}(P) \]

\[
\alpha(\uparrow\text{succ}(\gamma(\{E\}))) \\
= \alpha(\uparrow\text{succ}(\{n \mid \text{even}(n)\})) \\
= \alpha(\{\text{succ}(n) \mid \text{even}(n)\}) \\
= \alpha(\{n \mid \text{odd}(n)\}) \\
= \{0\} \\
\triangleq \uparrow\text{flip}(\{E\})
\]
Abstract Interpretation

$$\varphi(P) \mapsto \ldots \mapsto \varphi(P)$$

calc : $\alpha(\uparrow \text{succ}(\gamma(P))) = \ldots \triangleq \uparrow \text{flip}(P)$
Abstract Interpretation

\[ \mathcal{G}(P) \mapsto \ldots \mapsto \mathcal{G}(P) \]

\[
\text{calc} : \alpha(\uparrow\text{succ}(\gamma(P))) = \ldots \uparrow \triangleq \uparrow\text{flip}(P)
\]

\[ \mathcal{G}(P) = (P \rightarrow \text{prop}) \approx \text{"specification"} \]

\[ \mathcal{G}(P) = \{P\} \approx \text{"constructed"} \]
Abstract Interpretation

✓ Optimal specifications
✓ Calculational framework
✗ Requires axioms
✗ Definitions don’t compute
Four Stories

Direct Verification

Abstract Interpretation

Kleisli GCs

Constructive GCs

✗ calculate

✗ mechanize

✓ calculate

½ mechanize

✓ calculate

✓ mechanize
Kleisli GCs

\[ \alpha : \wp(\mathbb{N}) \rightarrow \wp(\mathbb{P}) \quad \text{\textasciitilde succ} : \wp(\mathbb{N}) \rightarrow \wp(\mathbb{N}) \]

\[ \gamma : \wp(\mathbb{P}) \rightarrow \wp(\mathbb{N}) \quad \text{\textasciitilde flip} : \wp(\mathbb{P}) \rightarrow \wp(\mathbb{P}) \]
Kleisli GCs

\[ \alpha : \mathbb{N} \to \mathcal{P}(\mathbb{P}) \quad \uparrow \text{succ} : \mathbb{N} \to \mathcal{P}(\mathbb{N}) \]

\[ \gamma : \mathbb{P} \to \mathcal{P}(\mathbb{N}) \quad \uparrow \text{flip} : \mathbb{P} \to \mathcal{P}(\mathbb{P}) \]
Kleisli GCs

\[ \alpha : \mathbb{N} \to \mathbb{P} \to \text{prop} \quad \uparrow\text{succ} : \mathbb{N} \to \mathbb{N} \to \text{prop} \]

\[ \gamma : \mathbb{P} \to \mathbb{N} \to \text{prop} \quad \uparrow\text{flip} : \mathbb{P} \to \mathbb{P} \to \text{prop} \]

\[ \mathcal{Φ}(X) = X \to \text{prop} \]
Kleisli GCs

\[ \alpha : \mathbb{N} \to \wp(\mathbb{P}) \]

\[ \gamma : \mathbb{P} \to \wp(\mathbb{N}) \]

\[ \uparrow \text{succ} : \mathbb{N} \to \wp(\mathbb{N}) \]

\[ \uparrow \text{flip} : \mathbb{P} \to \wp(\mathbb{P}) \]
Kleisli GCs

\[ \alpha : \mathbb{N} \to \mathcal{P}(\mathbb{P}) \]
\[ \gamma : \mathbb{P} \to \mathcal{P}(\mathbb{N}) \]
\[ \text{↑succ} : \mathbb{N} \to \mathcal{P}(\mathbb{N}) \]
\[ \text{↑flip} : \mathbb{P} \to \mathcal{P}(\mathbb{P}) \]

\[
N \subseteq \gamma(\alpha(N)) \land \alpha(\gamma(P)) \subseteq P
\]

\[
N \subseteq \gamma(P) \iff \alpha(N) \subseteq P
\]
Kleisli GCs

\[ \alpha : \mathbb{N} \to \mathcal{P}(\mathcal{P}) \quad \text{\uparrow succ} : \mathbb{N} \to \mathcal{P}(\mathbb{N}) \]
\[ \gamma : \mathcal{P} \to \mathcal{P}(\mathbb{N}) \quad \text{\uparrow flip} : \mathcal{P} \to \mathcal{P}(\mathcal{P}) \]

\[
\begin{align*}
id \subseteq \gamma \circ \alpha \quad \land \quad \alpha \circ \gamma \subseteq \text{id} \\
\text{id}(\mathbb{N}) \subseteq \gamma(\mathcal{P}) \iff \alpha(\mathbb{N}) \subseteq \text{id}(\mathcal{P})
\end{align*}
\]
Kleisli GCs

\( \alpha : \mathbb{N} \rightarrow \wp(\mathbb{P}) \) \hspace{1cm} \uparrow\text{succ} : \mathbb{N} \rightarrow \wp(\mathbb{N})

\( \gamma : \mathbb{P} \rightarrow \wp(\mathbb{N}) \) \hspace{1cm} \uparrow\text{flip} : \mathbb{P} \rightarrow \wp(\mathbb{P})

\[
\text{ret} \subseteq \gamma \circ \alpha \quad \land \quad \alpha \circ \gamma \subseteq \text{ret}
\]

\[
\text{ret}(n) \subseteq \gamma(p) \iff \alpha(n) \subseteq \text{ret}(p)
\]
Kleisli GCs

\[ \alpha : \mathbb{N} \to \wp(\mathbb{P}) \quad \uparrow\text{succ} : \mathbb{N} \to \wp(\mathbb{N}) \]

\[ \gamma : \mathbb{P} \to \wp(\mathbb{N}) \quad \uparrow\text{flip} : \mathbb{P} \to \wp(\mathbb{P}) \]

\[
\begin{array}{l}
\text{sound} : \alpha \circ \uparrow\text{succ} \circ \gamma \subseteq \uparrow\text{flip} \\
\end{array}
\]
Kleisli GCs

\[
\begin{align*}
\alpha & : \mathbb{N} \to \mathcal{P}(\mathcal{P}) \\
\gamma & : \mathcal{P} \to \mathcal{P}(\mathbb{N}) \\
\uparrow \text{succ} & : \mathbb{N} \to \mathcal{P}(\mathbb{N}) \\
\uparrow \text{flip} & : \mathcal{P} \to \mathcal{P}(\mathcal{P})
\end{align*}
\]

\[
\text{ret} \subseteq \gamma \ast \alpha \land \alpha \ast \gamma \subseteq \text{ret}
\]

\[
\text{ret}(n) \subseteq \gamma(p) \iff \alpha(n) \subseteq \text{ret}(p)
\]

sound : \(\alpha \ast \uparrow \text{succ} \ast \gamma \subseteq \uparrow \text{flip}\)
Kleisli GCs

✓ Optimal specifications

✓ Calculational framework

✓ No axioms

✗ Definitions don’t compute
Four Stories

Direct Verification ✓ calculate
Abstract Interpretation ✓ mechanize
Kleisli GCs ✓ calculate
½ mechanize
Constructive GCs ✓ calculate ✓ mechanize
Constructive GCs

\[ \alpha : \mathbb{N} \rightarrow \wp(\mathbb{P}) \]
\[ \gamma : \mathbb{P} \rightarrow \wp(\mathbb{N}) \]

\[ \text{ret} \sqsubseteq \gamma \otimes \alpha \]
\[ \alpha \otimes \gamma \sqsubseteq \text{ret} \]
Constructive GCs

\( \alpha : \mathbb{N} \rightarrow \wp(\mathbb{P}) \)

\( \gamma : \mathbb{P} \rightarrow \wp(\mathbb{N}) \)

\( \text{ret} \sqsubseteq \gamma \circ \alpha \)

\( \alpha \circ \gamma \sqsubseteq \text{ret} \)

---

\( \exists (\eta : \mathbb{N} \rightarrow \mathbb{P}). \ \alpha(x) = \text{ret}(\eta(x)) \)
Constructive GCs

\[ \alpha : \mathbb{N} \to \wp(\mathbb{P}) \]
\[ \gamma : \mathbb{P} \to \wp(\mathbb{N}) \]

\[ \text{ret} \subseteq \gamma \circ \alpha \]
\[ \alpha \circ \gamma \subseteq \text{ret} \]

\[ \exists (\eta : \mathbb{N} \to \mathbb{P}). \alpha(x) = \text{ret}(\eta(x)) \]
Constructive GCs

\[ \alpha : \mathbb{N} \rightarrow \wp (\mathbb{P}) \]

\[ \gamma : \mathbb{P} \rightarrow \wp (\mathbb{N}) \]
Constructive GCs

\[ \alpha : \mathbb{N} \to \mathbb{P} \]
\[ \gamma : \mathbb{P} \to \wp(\mathbb{N}) \]
Constructive GCs

parity : \mathbb{N} \rightarrow \mathbb{P}

[\_ ] : \mathbb{P} \rightarrow \wp(\mathbb{N})
Constructive GCs

\[
\text{parity} : \mathbb{N} \rightarrow \mathcal{P} \\
[\_] : \mathcal{P} \rightarrow \mathcal{P}(\mathbb{N})
\]

\[n \in \lfloor \text{parity}(n) \rfloor\]
Constructive GCs

\[
\text{parity} : \mathbb{N} \to \wp
\]

\[
[\_] : \wp \to \wp(\mathbb{N})
\]

\[
n \in [\text{parity}(n)] \land n \in [p] \Rightarrow \text{parity}(n) \subseteq p
\]
Constructive GCs

\[
\text{parity} : \mathbb{N} \rightarrow \mathcal{P} \\
\llbracket _\_ \rrbracket : \mathcal{P} \rightarrow \mathcal{P}(\mathbb{N})
\]

\[
n \in \llbracket \text{parity}(n) \rrbracket \land n \in \llbracket p \rrbracket \Rightarrow \text{parity}(n) \subseteq p
\]

\[
n \in \llbracket p \rrbracket \iff \text{parity}(n) \subseteq p
\]
Constructive GCs

\[
\text{parity} : \mathbb{N} \to \mathcal{P} \\
\mathcal{[\_]} : \mathcal{P} \to \wp(\mathbb{N})
\]

\[
n \in \mathcal{[\text{parity}(n)]} \land n \in \mathcal{[p]} \Rightarrow \text{parity}(n) \subseteq p
\]

\[
n \in \mathcal{[p]} \iff \text{parity}(n) \subseteq p
\]

\[
\text{sound} : n \in \mathcal{[p]} \Rightarrow \text{parity}(\text{succ}(n)) \subseteq \text{flip}(p)
\]
Constructive GCs

✓ Optimal specifications
✓ Calculational framework
✓ No Axioms
✓ Definitions that compute
And More
And More

- Metatheory complete w.r.t. subset of classical GC
And More

- Metatheory complete w.r.t. subset of classical GC
- Adjunction analogous to classical GCs
And More

• Metatheory complete w.r.t. subset of classical GC
• Adjunction analogous to classical GCs
• Case Study: Calculational AI [Cousot 1999]
And More

• Metatheory complete w.r.t. subset of classical GC

• Adjunction analogous to classical GCs

• Case Study: Calculational AI [Cousot 1999]

• Case Study: AGT [Garcia, Clark and Tanter 2016]
And More

• Metatheory complete w.r.t. subset of classical GC

• Adjunction analogous to classical GCs

• Case Study: Calculational AI  [Cousot 1999]

• Case Study: AGT  [Garcia, Clark and Tanter 2016]

• Sound, optimal and computable AIs by construction
And More

- Metatheory complete w.r.t. subset of classical GC
- Adjunction analogous to classical GCs
- Case Study: Calculational AI [Cousot 1999]
- Case Study: AGT [Garcia, Clark and Tanter 2016]
- Sound, optimal and computable AIs by construction
- Metatheory and case studies all verified in Agda
Constructive GCs

\[ \alpha : \mathbb{N} \to \mathcal{P} \]
\[ \gamma : \mathcal{P} \to \mathcal{P}(\mathbb{N}) \]

\[ n \in \gamma(p) \iff \alpha(n) \subseteq p \]

✓ calculate
✓ mechanize