Abstract Diagnosis
of First Order Functional Logic Programs

Giovanni Bacci Marco Comini

Dipartimento di Matematica e Informatica
University of Udine

LOPSTR 2010
23 July, Hagenberg

Motivations

incompleteness symptom

incorrectness symptom

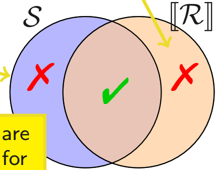
Automatic Debugging

Input: program $\mathcal{R}$ + specification $\mathcal{S}$

Task: automatically locate bugs in $\mathcal{R}$

in general it is **undecidable**

rules which are responsible for symptoms ✗



How to cope with this problem?

- Declarative Debugging $\Rightarrow$ partial inspection of the symptomatic *computation tree*
- **Abstract Diagnosis** $\Rightarrow$ use a correct approximation of the semantics which is finitely representable

Motivations

incompleteness symptom

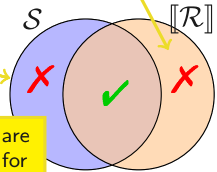
incorrectness symptom

Automatic Debugging

Input: program $\mathcal{R}$ + specification $\mathcal{S}$

Task: automatically locate bugs in $\mathcal{R}$

in general it is **undecidable**



rules which are responsible for symptoms ✗

How to cope with this problem?

+ Declarative Debugging
+ **Abstract Diagnosis**

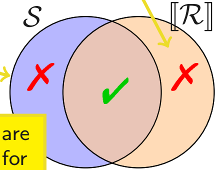
There are some cons:

+ symptom driven
+ semi-automatic
+ can't ensure that a property holds for P

Motivations

incompleteness symptom

incorrectness symptom



Automatic Debugging

Input: program $\mathcal{R}$ + specification $\mathcal{S}$

Task: automatically locate bugs in $\mathcal{R}$

in general it is **undecidable**

rules which are responsible for symptoms ✗

How to cope with this problem?

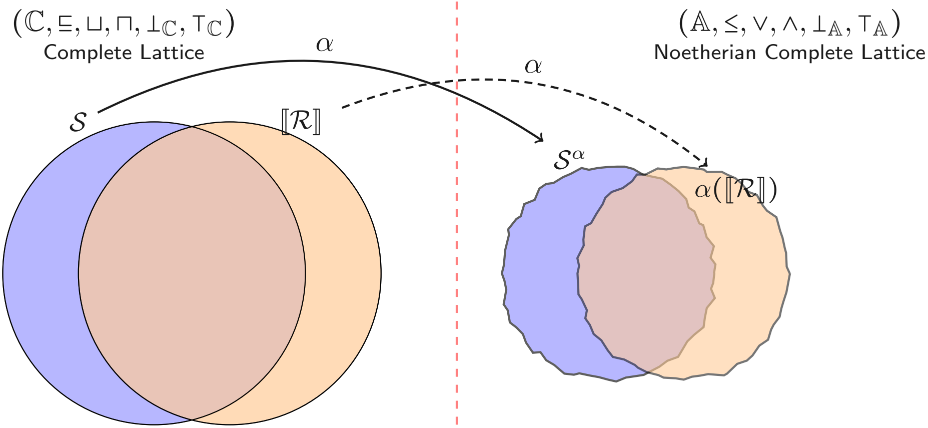
- Declarative Debugging
- **Abstract Diagnosis**

does not suffer

There are some cons:

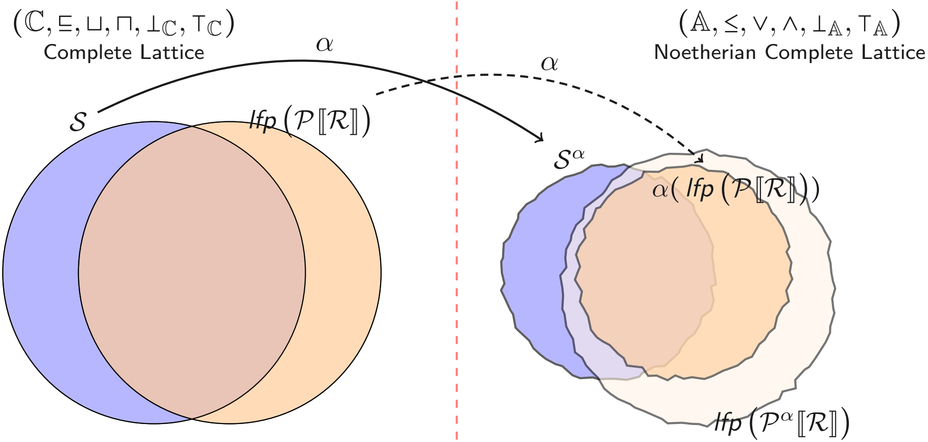
- symptom driven
- semi-automatic
- can't ensure that a property holds for P

The main idea



The Recipe:

+ Abstraction function $\alpha\colon \mathbb{C} \to \mathbb{A}$



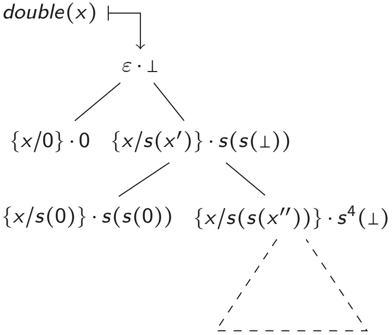
The Recipe:

- Abstraction function $\alpha : \mathbb{C} \to \mathbb{A}$
- Fix-point operator $\mathcal{P}[\![\mathcal{R}]\!] : \mathbb{C} \to \mathbb{C}$

Denotations: Incremental Answer Trees (Concrete Semantics)

R_1: $double(0) \to 0$

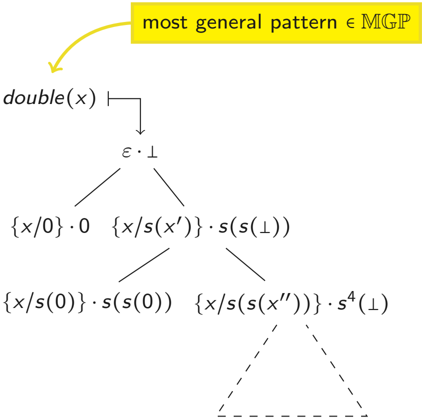
R_2: $double(s(x)) \to s(s(double(x)))$



Denotations: Incremental Answer Trees (Concrete Semantics)

R_1: $double(0) \to 0$

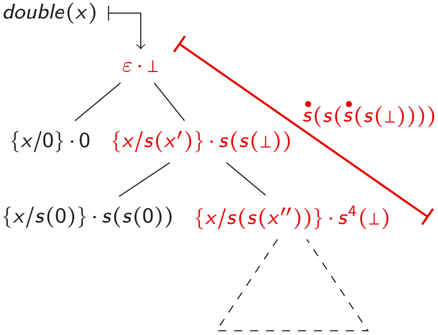
R_2: $double(s(x)) \to s(s(double(x)))$



Denotations: Incremental Answer Trees (Concrete Semantics)

R_1: $double(0) \to 0$

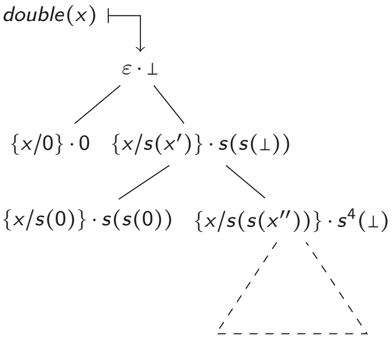
R_2: $double(s(x)) \to s(s(double(x)))$



R_1: $double(0) \to 0$

R_2: $double(s(x)) \to s(s(double(x)))$

the two denotations are isomorphic



$$\Big\{ \varepsilon \cdot \bot,\ \{x/0\} \cdot \dot{0},\ \{x/s(x')\} \cdot \dot{s}(s(\bot)),\ \{x/s(0)\} \cdot \dot{s}(s(\dot{0})),\ \{x/s(s(x'))\} \cdot \dot{s}(s(\dot{s}(s(\bot)))), \ldots \Big\}$$

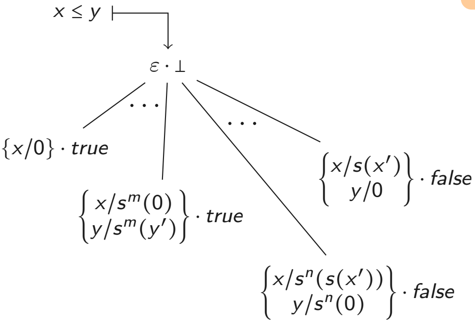
$\bullet \Longrightarrow$ some steps has taken place there

$R_1: 0 \leq y \to True$

$R_2: s(x) \leq 0 \to False$

$R_3: s(x) \leq s(y) \to x \leq y$

the two denotations are isomorphic



$$\Big\{ \varepsilon \cdot \bot, \{x/0\} \cdot \overset{\bullet}{true}, \ldots \\ \{x/s^i(0), y/s^i(y')\} \cdot \overset{\bullet}{true}, \ldots \\ \{x/s(x'), y/0\} \cdot \overset{\bullet}{true}, \ldots \\ \{x/s^{i+1}(x'), y/s^i(0)\} \cdot \overset{\bullet}{true}, \ldots \Big\}$$

$\bullet \Longrightarrow$ some steps has taken place there

Fix-point operator (Concrete Semantics)

$$\mathcal{P}[\![\mathcal{R}]\!]_{\mathcal{I}} := \lambda f(\vec{x}).\ \{\varepsilon \cdot \bot\} \cup \left\{ (\{\vec{x}/\vec{t}\}\sigma)\restriction_{\vec{x}} \cdot \dot{s} \;\middle|\; \begin{array}{l} f(\vec{t}) \to r \ll \mathcal{R}, \\ \sigma \cdot s \in \mathcal{E}[\![r]\!]_{\mathcal{I}},\ s \neq \bot \end{array} \right\}$$

where terms are evaluated by means of $\mathcal{E}[\![\ ]\!]_{\mathcal{I}}$

$$\mathcal{E}[\![x]\!]_{\mathcal{I}} := \{\varepsilon \cdot x\}$$

$$\mathcal{E}[\![f(\vec{t})]\!]_{\mathcal{I}} := \left\{ (\vartheta\eta)\restriction_{\vec{t}} \cdot r\eta \;\middle|\; \begin{array}{l} \sigma_i \cdot s_i \in \mathcal{E}[\![t_i]\!]_{\mathcal{I}} \text{ for } i = 1, \ldots, n \\ \vartheta = mgu(\sigma_1, \ldots, \sigma_n),\ \mu \cdot r \ll \mathcal{I}(f(\vec{x})) \\ \exists \eta = \mathring{mgu}_{Var(r)}(f(\vec{x})\mu, f(\vec{s})\vartheta) \end{array} \right\}$$

Fix-point operator (Concrete Semantics)

a rule is taken

$$\mathcal{P}[\![\mathcal{R}]\!]_{\mathcal{I}} := \lambda f(\vec{x}).\ \{\varepsilon \cdot \bot\} \cup \left\{ (\{\vec{x}/\vec{t}\}\sigma)\restriction_{\vec{x}} \cdot \overset{\bullet}{s} \;\middle|\; \begin{array}{l} f(\vec{t}) \to r \ll \mathcal{R}, \\ \sigma \cdot s \in \mathcal{E}[\![r]\!]_{\mathcal{I}},\ s \neq \bot \end{array} \right\}$$

where terms are evaluated by means of $\mathcal{E}[\![\ ]\!]_{\mathcal{I}}$

$$\mathcal{E}[\![x]\!]_{\mathcal{I}} := \{\varepsilon \cdot x\}$$

$$\mathcal{E}[\![f(\vec{t})]\!]_{\mathcal{I}} := \left\{ (\vartheta\eta)\restriction_{\vec{t}} \cdot r\eta \;\middle|\; \begin{array}{l} \sigma_i \cdot s_i \in \mathcal{E}[\![t_i]\!]_{\mathcal{I}} \text{ for } i = 1, \dots, n \\ \vartheta = mgu(\sigma_1, \dots, \sigma_n),\ \mu \cdot r \ll \mathcal{I}(f(\vec{x})) \\ \exists \eta = \mathring{mgu}_{Var(r)}(f(\vec{x})\mu, f(\vec{s})\vartheta) \end{array} \right\}$$

a rule is taken

$$\mathcal{P}[\![\mathcal{R}]\!]_{\mathcal{I}} := \lambda f(\vec{x}).\ \{\varepsilon \cdot \bot\} \cup \left\{ (\{\vec{x}/\vec{t}\}\sigma)\restriction_{\vec{x}} \cdot \dot{s} \;\middle|\; \begin{array}{l} f(\vec{t}) \to r \ll \mathcal{R}, \\ \sigma \cdot s \in \mathcal{E}[\![r]\!]_{\mathcal{I}},\ s \neq \bot \end{array} \right\}$$

its rhs is evaluated w.r.t. $\mathcal{I}$

where terms are evaluated by means of $\mathcal{E}[\![\ ]\!]_{\mathcal{I}}$

$$\mathcal{E}[\![x]\!]_{\mathcal{I}} := \{\varepsilon \cdot x\}$$

$$\mathcal{E}[\![f(\vec{t})]\!]_{\mathcal{I}} := \left\{ (\vartheta\eta)\restriction_{\vec{t}} \cdot r\eta \;\middle|\; \begin{array}{l} \sigma_i \cdot s_i \in \mathcal{E}[\![t_i]\!]_{\mathcal{I}} \text{ for } i = 1, \ldots, n \\ \vartheta = mgu(\sigma_1, \ldots, \sigma_n),\ \mu \cdot r \ll \mathcal{I}(f(\vec{x})) \\ \exists \eta = \mathring{mgu}_{Var(r)}(f(\vec{x})\mu, f(\vec{s})\vartheta) \end{array} \right\}$$

the new contribution is annotated

a rule is taken

$$\mathcal{P}[\![\mathcal{R}]\!]_{\mathcal{I}} := \lambda f(\vec{x}).\ \{\varepsilon \cdot \bot\} \cup \left\{ (\{\vec{x}/\vec{t}\}\sigma)\restriction_{\vec{x}} \cdot \overset{\bullet}{s} \;\middle|\; \begin{array}{l} f(\vec{t}) \to r \ll \mathcal{R}, \\ \sigma \cdot s \in \mathcal{E}[\![r]\!]_{\mathcal{I}},\ s \neq \bot \end{array} \right\}$$

its rhs is evaluated w.r.t. $\mathcal{I}$

where terms are evaluated by means of $\mathcal{E}[\![\ ]\!]_{\mathcal{I}}$

$$\mathcal{E}[\![x]\!]_{\mathcal{I}} := \{\varepsilon \cdot x\}$$

$$\mathcal{E}[\![f(\vec{t})]\!]_{\mathcal{I}} := \left\{ (\vartheta\eta)\restriction_{\vec{t}} \cdot r\eta \;\middle|\; \begin{array}{l} \sigma_i \cdot s_i \in \mathcal{E}[\![t_i]\!]_{\mathcal{I}} \text{ for } i = 1, \ldots, n \\ \vartheta = mgu(\sigma_1, \ldots, \sigma_n),\ \mu \cdot r \ll \mathcal{I}(f(\vec{x})) \\ \exists \eta = m\mathring{g}u_{Var(r)}(f(\vec{x})\mu, f(\vec{s})\vartheta) \end{array} \right\}$$

the new contribution is annotated

a rule is taken

$$\mathcal{P}[\![\mathcal{R}]\!]_{\mathcal{I}} := \lambda f(\vec{x}).\ \{\varepsilon \cdot \bot\} \cup \left\{ (\{\vec{x}/\vec{t}\}\sigma)\restriction_{\vec{x}} \cdot \dot{\mathbf{s}} \;\middle|\; \begin{array}{l} f(\vec{t}) \to r \ll \mathcal{R}, \\ \sigma \cdot s \in \mathcal{E}[\![r]\!]_{\mathcal{I}},\ s \neq \bot \end{array} \right\}$$

its rhs is evaluated w.r.t. $\mathcal{I}$

where terms are evaluated by means of $\mathcal{E}[\![\ ]\!]_{\mathcal{I}}$

$$\mathcal{E}[\![x]\!]_{\mathcal{I}} := \{\varepsilon \cdot x\}$$

$$\mathcal{E}[\![f(\vec{t})]\!]_{\mathcal{I}} := \left\{ (\vartheta\eta)\restriction_{\vec{t}} \cdot r\eta \;\middle|\; \begin{array}{l} \sigma_i \cdot s_i \in \mathcal{E}[\![t_i]\!]_{\mathcal{I}} \text{ for } i = 1, \dots, n \\ \vartheta = mgu(\sigma_1, \dots, \sigma_n),\ \mu \cdot r \ll \mathcal{I}(f(\vec{x})) \\ \exists \eta = \mathring{mgu}_{Var(r)}(f(\vec{x})\mu, f(\vec{s})\vartheta) \end{array} \right\}$$

Fix-point operator

$$\mathcal{P}[\![\mathcal{R}]\!]_{\mathcal{I}} := \lambda f(\vec{x}).\ \{\varepsilon \cdot \bot\} \cup \left\{ (\{\vec{x}/\vec{t}\}\sigma){\restriction_{\vec{x}}} \cdot \dot{s} \;\middle|\; \begin{array}{l} f(\vec{t}) \to r \ll \mathcal{R}, \\ \sigma \cdot s \in \mathcal{E}[\![r]\!]_{\mathcal{I}},\ s \neq \bot \end{array} \right\}$$

where terms are evaluated by means of $\mathcal{E}[\![\ ]\!]_{\mathcal{I}}$

$$\mathcal{E}[\![x]\!]_{\mathcal{I}} := \{\varepsilon \cdot x\}$$

$$\mathcal{E}[\![f(\vec{t})]\!]_{\mathcal{I}} := \left\{ (\vartheta\eta){\restriction_{\vec{t}}} \cdot r\eta \;\middle|\; \begin{array}{l} \sigma_i \cdot s_i \in \mathcal{E}[\![t_i]\!]_{\mathcal{I}} \text{ for } i = 1, \dots, n \\ \vartheta = mgu(\sigma_1, \dots, \sigma_n),\ \mu \cdot r \ll \mathcal{I}(f(\vec{x})) \\ \exists \eta = \mathring{mgu}_{Var(r)}(f(\vec{x})\mu, f(\vec{s})\vartheta) \end{array} \right\}$$

Fix-point operator

$$\mathcal{P}[\![\mathcal{R}]\!]_{\mathcal{I}} := \lambda f(\vec{x}).\ \{\varepsilon \cdot \bot\} \cup \left\{ (\{\vec{x}/\vec{t}\}\sigma)\restriction_{\vec{x}} \cdot \overset{\bullet}{s} \;\middle|\; \begin{array}{l} f(\vec{t}) \to r \ll \mathcal{R}, \\ \sigma \cdot s \in \mathcal{E}[\![r]\!]_{\mathcal{I}},\ s \neq \bot \end{array} \right\}$$

where terms are eva … of $\mathcal{E}[\![\ ]\!]_{\mathcal{I}}$

sub-components are evaluated in parallel

$$\mathcal{E}[\![x]\!]_{\mathcal{I}} := \{\varepsilon \cdot x\}$$

$$\mathcal{E}[\![f(\vec{t})]\!]_{\mathcal{I}} := \left\{ (\vartheta\eta)\restriction_{\vec{t}} \cdot r\eta \;\middle|\; \begin{array}{l} \sigma_i \cdot s_i \in \mathcal{E}[\![t_i]\!]_{\mathcal{I}} \text{ for } i = 1, \ldots, n \\ \vartheta = mgu(\sigma_1, \ldots, \sigma_n),\ \mu \cdot r \ll \mathcal{I}(f(\vec{x})) \\ \exists \eta = \mathring{mgu}_{Var(r)}(f(\vec{x})\mu, f(\vec{s})\vartheta) \end{array} \right\}$$

$$\mathcal{P}[\![\mathcal{R}]\!]_{\mathcal{I}} := \lambda f(\vec{x}).\ \{\varepsilon \cdot \bot\} \cup \left\{ (\{\vec{x}/\vec{t}\}\sigma){\restriction_{\vec{x}}} \mathbin{\dot{\cdot}} s \;\middle|\; \begin{array}{l} f(\vec{t}) \to r \ll \mathcal{R}, \\ \sigma \cdot s \in \mathcal{E}[\![r]\!]_{\mathcal{I}},\ s \neq \bot \end{array} \right\}$$

where terms are eva of $\mathcal{E}[\![\ ]\!]_{\mathcal{I}}$

sub-components are evaluated in parallel

$$\mathcal{E}[\![x]\!]_{\mathcal{I}} := \{\varepsilon \cdot x\}$$

$$\mathcal{E}[\![f(\vec{t})]\!]_{\mathcal{I}} := \left\{ (\vartheta\eta){\restriction_{\vec{t}}} \cdot r\eta \;\middle|\; \begin{array}{l} \sigma_i \cdot s_i \in \mathcal{E}[\![t_i]\!]_{\mathcal{I}} \text{ for } i = 1, \ldots, n \\ \vartheta = mgu(\sigma_1, \ldots, \sigma_n),\ \mu \cdot r \ll \mathcal{I}(f(\vec{x})) \\ \exists \eta = \mathring{mgu}_{Var(r)}(f(\vec{x})\mu, f(\vec{s})\vartheta) \end{array} \right\}$$

partial answer as $f(\vec{x})\mu \to^* r$

Fix-point operator (Concrete Semantics)

$$\mathcal{P}[\![\mathcal{R}]\!]_{\mathcal{I}} := \lambda f(\vec{x}).\ \{\varepsilon \cdot \bot\} \cup \left\{ (\{\vec{x}/\vec{t}\}\sigma)\restriction_{\vec{x}} \cdot \dot{s} \;\middle|\; \begin{array}{l} f(\vec{t}) \to r \ll \mathcal{R}, \\ \sigma \cdot s \in \mathcal{E}[\![r]\!]_{\mathcal{I}},\ s \neq \bot \end{array} \right\}$$

where terms are eva … of $\mathcal{E}[\![\ ]\!]_{\mathcal{I}}$

$$\mathcal{E}[\![x]\!]_{\mathcal{I}} := \{\varepsilon \cdot x\}$$

$$\mathcal{E}[\![f(\vec{t})]\!]_{\mathcal{I}} := \left\{ (\vartheta\eta)\restriction_{\vec{t}} \cdot r\eta \;\middle|\; \begin{array}{l} \sigma_i \cdot s_i \in \mathcal{E}[\![t_i]\!]_{\mathcal{I}} \text{ for } i = 1, \dots, n \\ \vartheta = mgu(\sigma_1, \dots, \sigma_n),\ \mu \cdot r \ll \mathcal{I}(f(\vec{x})) \\ \exists \eta = \mathring{mgu}_{Var(r)}(f(\vec{x})\mu, f(\vec{s})\vartheta) \end{array} \right\}$$

sub-components are evaluated in parallel

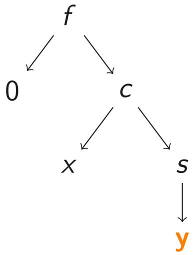
$\exists$ mgu + neededness

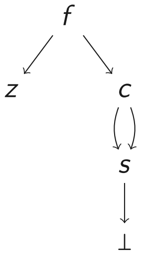
partial answer as $f(\vec{x})\mu \to^* r$

$m\mathring{g}u_{Var(r)}(f(\vec{x})\mu, f(\vec{s})\vartheta)$ **[Case 1]**

Rule taken from $\mathcal{I}$:
$f(0, c(x, s(\mathbf{y}))) \to^* c(x, \bot)$

Components partial eveluation:
$f(z, c(s(\bot), s(\bot)))$

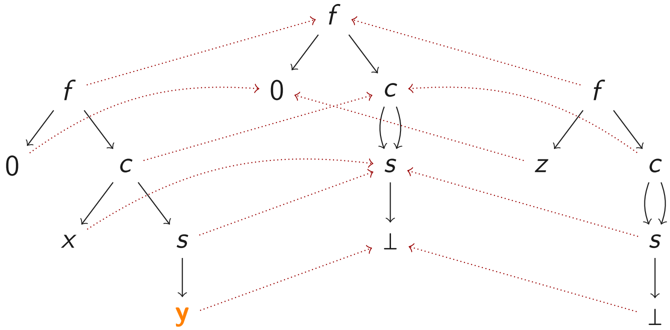




$\mathring{mgu}_{Var(r)}(f(\vec{x})\mu, f(\vec{s})\vartheta)$ **[Case 1]**

Rule taken from $\mathcal{I}$:
$f(0, c(x, s(\mathbf{y}))) \to^* c(x, \bot)$

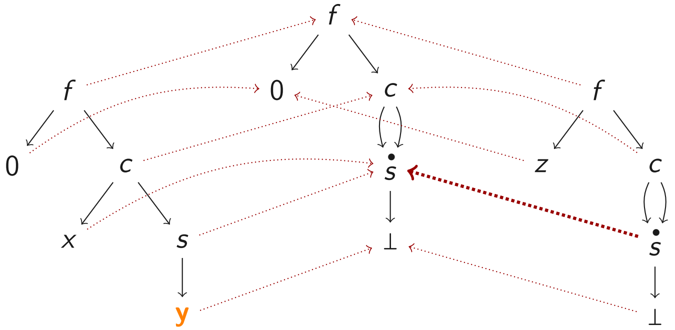
Components partial eveluation:
$f(z, c(s(\bot), s(\bot)))$



$\mathring{mgu}_{Var(r)}(f(\vec{x})\mu, f(\vec{s})\vartheta)$ **[Case 2]**

Rule taken from $\mathcal{I}$:
$f(0, c(x, s(\mathbf{y}))) \to^* c(x, \bot)$

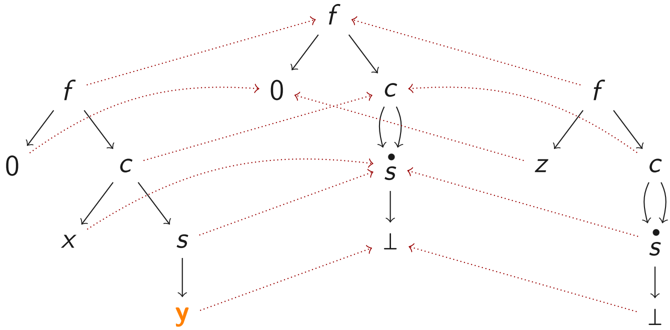
Components partial eveluation:
$f(z, c(\dot{s}(\bot), \dot{s}(\bot)))$



$\mathring{mgu}_{Var(r)}(f(\vec{x})\mu, f(\vec{s})\vartheta)$ **[Case 2]**

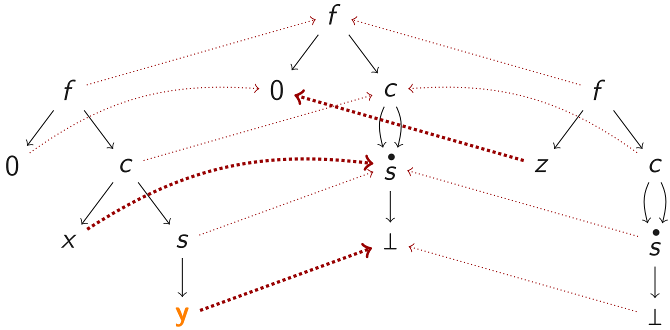
Rule taken from $\mathcal{I}$:
$f(0, c(x, s(\mathbf{y}))) \to^* c(x, \bot)$

Components partial eveluation:
$f(z, c(\dot{s}(\bot), \dot{s}(\bot)))$



Rule taken from $\mathcal{I}$:
$f(0, c(x, s(\mathbf{y}))) \to^* c(x, \bot)$

Components partial eveluation:
$f(z, c(\dot{s}(\bot), \dot{s}(\bot)))$



$\eta = \{x/\dot{s}(\bot), y/\bot, z/0\}$ (induced substitution)

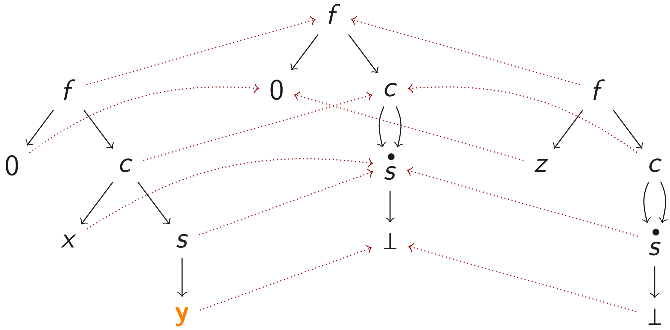
$\mathring{mgu}_{Var(r)}(f(\vec{x})\mu, f(\vec{s})\vartheta)$ **[Case 2]**

Rule taken from $\mathcal{I}$:

$f(0, c(x, s(\mathbf{y}))) \to^* c(x, \bot)$

Components partial eveluation:

$f(z, c(\overset{\bullet}{s}(\bot), \overset{\bullet}{s}(\bot)))$



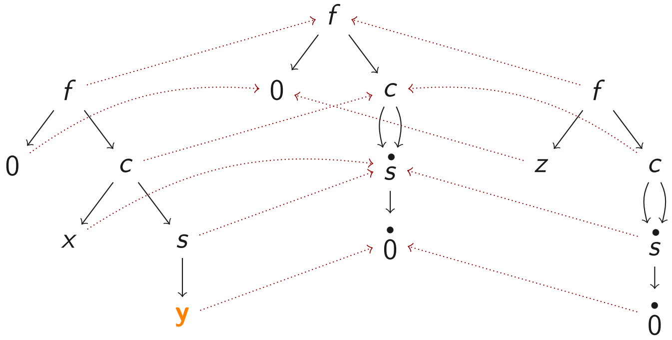
$m\mathring{g}u_{Var(r)}(f(\vec{x})\mu, f(\vec{s})\vartheta)$ **[Case 3]**

Rule taken from $\mathcal{I}$:

$f(0, c(x, s(\mathbf{y}))) \to^* c(x, \bot)$

Components partial eveluation:

$f(z, c(\dot{s}(\dot{0}), \dot{s}(\dot{0})))$

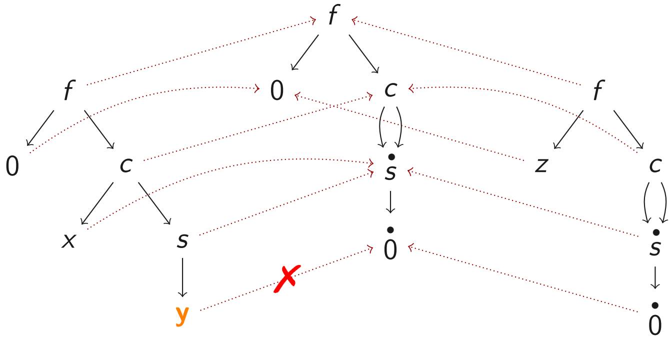


Rule taken from $\mathcal{I}$:

$f(0, c(x, s(\mathbf{y}))) \to^* c(x, \bot)$

Components partial eveluation:

$f(z, c(\dot{s}(\dot{0}), \dot{s}(\dot{0})))$



Fix-point characterization (Concrete Semantics)

$\mathcal{P}[\![\mathcal{R}]\!]$ is continuous $\Longrightarrow$

$$\mathcal{F}[\![\mathcal{R}]\!] := lfp\left(\mathcal{P}[\![\mathcal{R}]\!]\right) = \mathcal{P}[\![\mathcal{R}]\!]\uparrow\omega$$

Theorem (Correctness & Complete

it generates only correct partial answers

1. $\sigma \cdot s_{\perp} \in \mathcal{E}[\![t]\!]_{\mathcal{F}[\![\mathcal{R}]\!]} \Longrightarrow \exists s.\ t \overset{\sigma}{\underset{\mathcal{R}}{\rightsquigarrow}}{}^{*}\ s \text{ and } \tau_{\perp}(s) = s_{\perp}$
2. $t \overset{\sigma}{\underset{\mathcal{R}}{\rightsquigarrow}}{}^{*}\ s \Longrightarrow \exists \vartheta \leq \sigma. \exists s_{\perp} = \tau_{\perp}(s)\vartheta \text{ s.t. } \vartheta \cdot s_{\perp} \in \mathcal{E}[\![t]\!]_{\mathcal{F}[\![\mathcal{R}]\!]}$

it is able to generate every partial answers up to instantiation

Fix-point characterization (Concrete Semantics)

$\mathcal{P}[\![\mathcal{R}]\!]$ is continuous $\Longrightarrow$

$$\mathcal{F}[\![\mathcal{R}]\!] := lfp\left(\mathcal{P}[\![\mathcal{R}]\!]\right) = \mathcal{P}[\![\mathcal{R}]\!]\uparrow\omega$$

Theorem (Correctness & Complete

it generates only correct partial answers

1. $\sigma \cdot s_\perp \in \mathcal{E}[\![t]\!]_{\mathcal{F}[\![\mathcal{R}]\!]} \Longrightarrow \exists s.\ t \overset{\sigma}{\underset{\mathcal{R}}{\rightsquigarrow}}{}^{*} s$ and $\tau_\perp(s) = s_\perp$
2. $t \overset{\sigma}{\underset{\mathcal{R}}{\rightsquigarrow}}{}^{*} s \Longrightarrow \exists\vartheta \leq \sigma.\ \exists s_\perp = \tau_\perp(s)\vartheta$ s.t. $\vartheta \cdot s_\perp \in \mathcal{E}[\![t]\!]_{\mathcal{F}[\![\mathcal{R}]\!]}$

it is able to generate every partial answers up to instantiation

Fix-point characterization (Concrete Semantics)

$\mathcal{P}[\![\mathcal{R}]\!]$ is continuous $\Longrightarrow$

$$\mathcal{F}[\![\mathcal{R}]\!] := \mathit{lfp}\left(\mathcal{P}[\![\mathcal{R}]\!]\right) = \mathcal{P}[\![\mathcal{R}]\!]{\uparrow}\omega$$

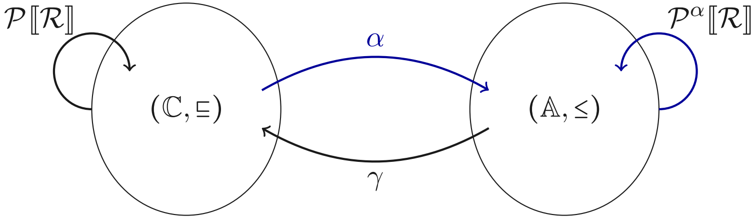
Theorem (Correctness & Complete...)

it generates only correct partial answers

1. $\sigma \cdot s_\perp \in \mathcal{E}[\![t]\!]_{\mathcal{F}[\![\mathcal{R}]\!]} \Longrightarrow \exists s.\ t \overset{\sigma}{\underset{\mathcal{R}}{\rightsquigarrow}}{}^{*} s \text{ and } \tau_\perp(s) = s_\perp$
2. $t \overset{\sigma}{\underset{\mathcal{R}}{\rightsquigarrow}}{}^{*} s \Longrightarrow \exists \vartheta \leq \sigma.\ \exists s_\perp = \tau_\perp(s)\vartheta \text{ s.t. } \vartheta \cdot s_\perp \in \mathcal{E}[\![t]\!]_{\mathcal{F}[\![\mathcal{R}]\!]}$

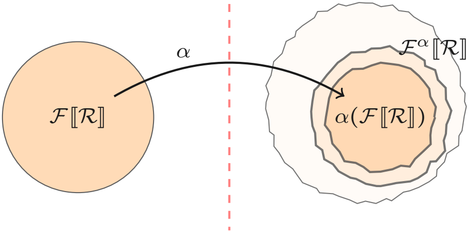
it is able to generate every partial answers up to instantiation

Inducing the abstract operator [Cousot 77]

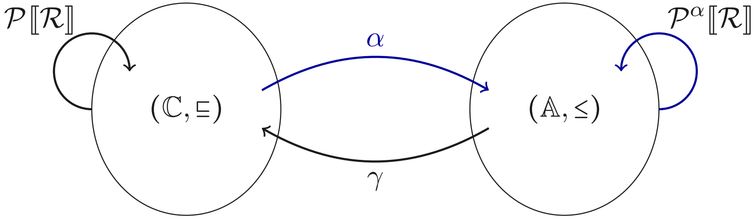


Results from the A.I. theory:

+ $\mathcal{P}^{\alpha}[\![\mathcal{R}]\!] := \alpha \circ \mathcal{P}[\![\mathcal{R}]\!] \circ \gamma$
+ $\mathcal{F}^{\alpha}[\![\mathcal{R}]\!] := \mathcal{P}^{\alpha}[\![\mathcal{R}]\!]\!\uparrow\!\omega$
+ $\alpha(\mathcal{F}^{\alpha}[\![\mathcal{R}]\!]) \leq \mathcal{F}^{\alpha}[\![\mathcal{R}]\!]$



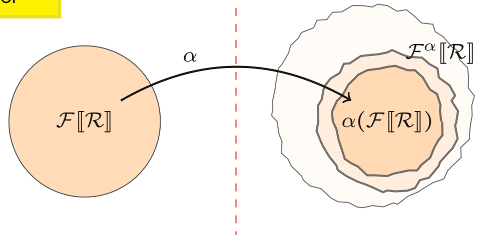
Inducing the abstract operator [Cousot 77]



Optimal abstract
fix-point operator

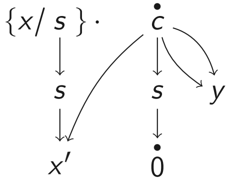
Results from the A.I. theory:

+ $\mathcal{P}^{\alpha}[\![\mathcal{R}]\!] := \alpha \circ \mathcal{P}[\![\mathcal{R}]\!] \circ \gamma$
+ $\mathcal{F}^{\alpha}[\![\mathcal{R}]\!] := \mathcal{P}^{\alpha}[\![\mathcal{R}]\!]{\uparrow}\omega$
+ $\alpha(\mathcal{F}^{\alpha}[\![\mathcal{R}]\!]) \leq \mathcal{F}^{\alpha}[\![\mathcal{R}]\!]$



Observed Property: it is observed the concrete behavior up to a fixed depth – k

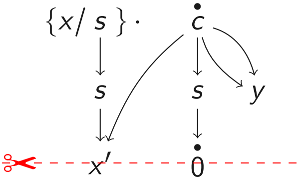
Abstract Domain: **depth(k)** (answers with depth at most k)



Case Study: depth(k) (Abstract Diagnosis)

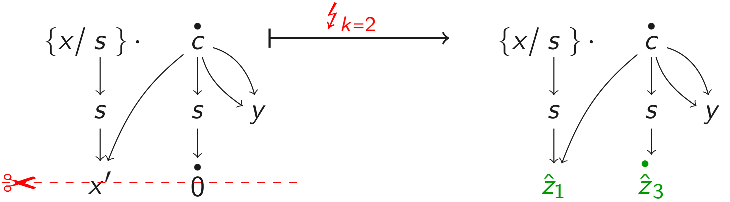
Observed Property: it is observed the concrete behavior up to a fixed depth – k

Abstract Domain: **depth(k)** (answers with depth at most k)



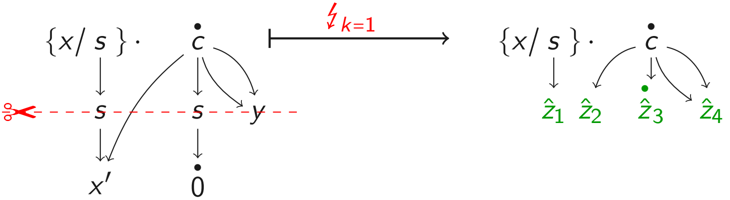
Observed Property: it is observed the concrete behavior up to a fixed depth – k

Abstract Domain: **depth(k)** (answers with depth at most k)



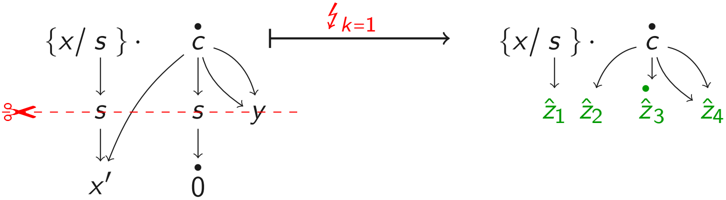
Observed Property: it is observed the concrete behavior up to a fixed depth – k

Abstract Domain: **depth(k)** (answers with depth at most k)



Observed Property: it is observed the concrete behavior up to a fixed depth – k

Abstract Domain: **depth(k)** (answers with depth at most k)



The abstraction over interpretations α^{κ} is obtained through successive extensions

Optimal abstract fix-point operator

$$
\begin{aligned}
\mathcal{P}^{\kappa}[\![\mathcal{R}]\!]_{\mathcal{I}^{\kappa}} &:= \alpha^{\kappa}(\mathcal{P}[\![\mathcal{R}]\!]_{\gamma^{\kappa}(\mathcal{I}^{\kappa})}) \\
&= \lambda f(\vec{x}).\ \{\varepsilon \cdot \bot\} \vee \bigvee \left\{ ((\{\vec{x}/\vec{t}\}\sigma)\upharpoonright_{\vec{x}} \cdot \overset{\bullet}{s})\lightning_{k} \;\middle|\; \begin{array}{l} f(\vec{t}) \to r \ll \mathcal{R}, \\ \sigma \cdot s \in \mathcal{E}^{\kappa}[\![r]\!]_{\mathcal{I}^{\kappa}},\ s \neq \bot \end{array} \right\}
\end{aligned}
$$

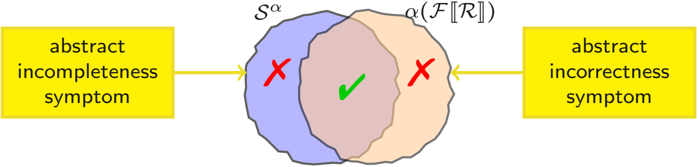
where

$$
\begin{aligned}
\mathcal{E}^{\kappa}[\![x]\!]_{\mathcal{I}^{\kappa}} &:= \{\varepsilon \cdot x\} \\
\mathcal{E}^{\kappa}[\![f(\vec{t})]\!]_{\mathcal{I}^{\kappa}} &:= \left\{ (\vartheta\eta)\upharpoonright_{\vec{t}} \cdot r\eta \;\middle|\; \begin{array}{l} \sigma_i \cdot s_i \in \mathcal{E}^{\kappa}[\![t_i]\!]_{I^{\kappa}} \text{ for } i = 1, \dots, n \\ \vartheta = mgu(\sigma_1, \dots, \sigma_n),\ \mu \cdot r \ll \mathcal{I}^{\kappa}(f(\vec{x})) \\ \exists \eta = m\mathring{g}u_{Var(r) \cup \widehat{\mathcal{V}}}(f(\vec{x})\mu, f(\vec{s})\vartheta) \end{array} \right\}
\end{aligned}
$$

Abstract Bugs & Symptoms

Let $\mathcal{R}$ be a program and α a property

- (abstract) **partially correct** w.r.t. $\mathcal{S}^\alpha$: $\alpha(\mathcal{F}[\![\mathcal{R}]\!]) \leq \mathcal{S}^\alpha$
- (abstract) **complete** w.r.t. $\mathcal{S}^\alpha$: $\mathcal{S}^\alpha \leq \alpha(\mathcal{F}[\![\mathcal{R}]\!])$



Task: automatically locate bugs responsible for symptoms

Problem: **interference** between incorrectness and uncovered errors **can be symptomless**

$\Downarrow$

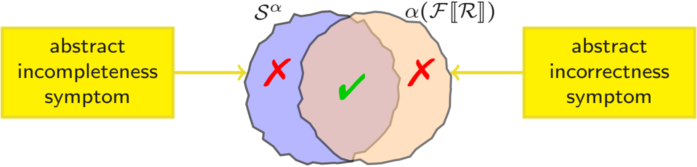
Declarative Diagnosis

cannot reveal all errors **symultaneosly**

Abstract Bugs & Symptoms (Abstract Diagnosis)

Let $\mathcal{R}$ be a program and α a property

- (abstract) **partially correct** w.r.t. $\mathcal{S}^\alpha$: $\alpha(\mathcal{F}[\![\mathcal{R}]\!]) \leq \mathcal{S}^\alpha$
- (abstract) **complete** w.r.t. $\mathcal{S}^\alpha$: $\mathcal{S}^\alpha \leq \alpha(\mathcal{F}[\![\mathcal{R}]\!])$



Task: automatically locate bugs responsible for symptoms

Problem: **interference** between incorrectness and uncovered errors **can be symptomless**

$\Downarrow$

Declarative Diagnosis **cannot** reveal all errors **symultaneosly**

Abstract Diagnosis Framework

Based on abstract version of Park's Induction Principle:

$$\mathcal{P}^{\alpha}[\![\mathcal{R}]\!]_{\mathcal{S}^{\alpha}} \stackrel{?}{\leq} \mathcal{S}^{\alpha}$$

- $e \leq \mathcal{P}^{\alpha}[\![\{l \to r\}]\!]_{\mathcal{S}^{\alpha}}$ and $e \nleq \mathcal{S}^{\alpha}$ (abstractly incorrect rule)

- $e \wedge \mathcal{P}^{\alpha}[\![\mathcal{R}]\!]_{\mathcal{S}^{\alpha}} = \bot_{\mathbb{A}}$ and $e \leq \mathcal{S}^{\alpha}$ (abstractly uncovered elem.)

Abstract Diagnosis Framework

Based on abstract version of Park's Induction Principle:

$$\mathcal{P}^{\alpha}[\![\mathcal{R}]\!]_{\mathcal{S}^{\alpha}} \overset{?}{\leq} \mathcal{S}^{\alpha}$$

using $\mathcal{S}^{\alpha}$, $l \to r$ produces e...

+ $e \leq \mathcal{P}^{\alpha}[\![\{l \to r\}]\!]_{\mathcal{S}^{\alpha}}$ and $e \nleq \mathcal{S}^{\alpha}$ (abstractly incorrect rule)

+ $e \wedge \mathcal{P}^{\alpha}[\![\mathcal{R}]\!]_{\mathcal{S}^{\alpha}} = \bot_{\mathbb{A}}$ and $e \leq \mathcal{S}^{\alpha}$ (abstractly uncovered elem.)

Abstract Diagnosis Framework

Based on abstract version of Park's Induction Principle:

using $\mathcal{S}^\alpha$, $l \to r$ produces e...

...but e was not expected by $\mathcal{S}^\alpha$

$$\ldots_{\alpha} \overset{?}{\leq} \mathcal{S}^\alpha$$

+ $e \leq \mathcal{P}^\alpha[\![\{l \to r\}]\!]_{\mathcal{S}^\alpha}$ and $e \not\leq \mathcal{S}^\alpha$ (abstractly incorrect rule)

+ $e \wedge \mathcal{P}^\alpha[\![\mathcal{R}]\!]_{\mathcal{S}^\alpha} = \bot_{\mathbb{A}}$ and $e \leq \mathcal{S}^\alpha$ (abstractly uncovered elem.)

Abstract Diagnosis Framework

Based on abstract version of Park's Induction Principle:

using $\mathcal{S}^\alpha$, $l \to r$ produces e...

...but e was not expected by $\mathcal{S}^\alpha$

$$\cdots_{\alpha} \overset{?}{\leq} \mathcal{S}^\alpha$$

+ $e \leq \mathcal{P}^\alpha[\![\{l \to r\}]\!]_{\mathcal{S}^\alpha}$ and $e \nleq \mathcal{S}^\alpha$ (abstractly incorrect rule)

using $\mathcal{S}^\alpha$, $\mathcal{R}$ can't produce e...

+ $e \wedge \mathcal{P}^\alpha[\![\mathcal{R}]\!]_{\mathcal{S}^\alpha} = \bot_{\mathbb{A}}$ and $e \leq \mathcal{S}^\alpha$ (abstractly uncovered elem.)

Abstract Diagnosis Framework

Based on abstract version of Park's Induction Principle:

using $\mathcal{S}^\alpha$, $l \to r$ produces $e\dots$

…but e was not expected by $\mathcal{S}^\alpha$

$\alpha \overset{?}{\leq} \mathcal{S}^\alpha$

+ $e \leq \mathcal{P}^\alpha[\![\{l \to r\}]\!]_{\mathcal{S}^\alpha}$ and $e \nleq \mathcal{S}^\alpha$ (abstractly incorrect rule)

using $\mathcal{S}^\alpha$, $\mathcal{R}$ can't produce $e\dots$

…but e was expected by $\mathcal{S}^\alpha$

+ $e \wedge \mathcal{P}^\alpha[\![\mathcal{R}]\!]_{\mathcal{S}^\alpha} = \bot_{\mathbb{A}}$ and $e \leq \mathcal{S}^\alpha$ (abstractly uncovered elem.)

Abstract Diagnosis Framework

Based on abstract version of Park's Induction Principle:

$\ldots{}^{\alpha} \overset{?}{\leq} \mathcal{S}^{\alpha}$

using $\mathcal{S}^{\alpha}$, $l \to r$ produces e...

...but e was not expected by $\mathcal{S}^{\alpha}$

+ $e \leq \mathcal{P}^{\alpha}[\![\{l \to r\}]\!]_{\mathcal{S}^{\alpha}}$ and $e \not\leq \mathcal{S}^{\alpha}$ (abstractly incorrect rule)

using $\mathcal{S}^{\alpha}$, $\mathcal{R}$ can't produce e...

...but e was expected by $\mathcal{S}^{\alpha}$

+ $e \wedge \mathcal{P}^{\alpha}[\![\mathcal{R}]\!]_{\mathcal{S}^{\alpha}} = \bot_{\mathbb{A}}$ and $e \leq \mathcal{S}^{\alpha}$ (abstractly uncovered elem.)

Pros:
+ Static test (requires just one $\mathcal{P}^{\alpha}[\![\mathcal{R}]\!]$ step on $\mathcal{S}^{\alpha}$)
+ reveal all abstract errors regardless of symptoms interference

Cons:
+ imprecision of α can lead to false positives:

$$\mathcal{P}[\![\{l \to r\}]\!]_{\mathcal{S}} \sqsubseteq \mathcal{S} \wedge \mathcal{P}^{\alpha}[\![\{l \to r\}]\!]_{\alpha(\mathcal{S})} \not\leq \alpha(\mathcal{S})$$

However

$$\mathcal{P}[\![\{l \to r\}]\!]_{\mathcal{S}} \not\sqsubseteq \mathcal{S} \wedge \mathcal{P}^{\alpha}[\![\{l \to r\}]\!]_{\alpha(\mathcal{S})} \not\leq \alpha(\mathcal{S}) \Longrightarrow r \text{ is abstractly incorrect}$$

Example: infinite computation (Abstract Diagnosis)

+ **Program:**

$$R\colon from(n) \to n : from(n)$$

+ **Specification:** with $\kappa = 3$

$$\mathcal{S}^{\kappa} := \begin{cases} from(n) \mapsto \{\varepsilon \cdot \bot,\ \varepsilon \cdot n \stackrel{\bullet}{:} \bot,\ \varepsilon \cdot n \stackrel{\bullet}{:} s(\hat{x}_1) \stackrel{\bullet}{:} \bot, \\ \qquad \varepsilon \cdot n \stackrel{\bullet}{:} s(\hat{x}_1) \stackrel{\bullet}{:} \hat{x}_2 \stackrel{\bullet}{:} \hat{x}_3,\ \varepsilon \cdot n \stackrel{\bullet}{:} s(\hat{x}_1) \stackrel{\bullet}{:} \hat{x}_2 \stackrel{\bullet}{:} \dot{\hat{x}}_3\} \end{cases}$$

We detect that rule R is abstractly incorrect since

$$\mathcal{P}^{\kappa}[\![\{R\}]\!]_{\mathcal{S}^{\kappa}} = \begin{cases} from(n) \mapsto \{\varepsilon \cdot \bot,\ \varepsilon \cdot n \stackrel{\bullet}{:} \bot,\ \varepsilon \cdot n \stackrel{\bullet}{:} n \stackrel{\bullet}{:} \bot, \\ \qquad \varepsilon \cdot n \stackrel{\bullet}{:} n \stackrel{\bullet}{:} \hat{x}_2 \stackrel{\bullet}{:} \hat{x}_3,\ \varepsilon \cdot n \stackrel{\bullet}{:} n \stackrel{\bullet}{:} \hat{x}_2 \stackrel{\bullet}{:} \dot{\hat{x}}_3\} \end{cases} \not\leq \mathcal{S}^{\kappa}$$

In declarative debugging the goal must be wrapped with an unneeded function (e.g. $take : Int \to [a] \to [a]$) in order to make the Computation Tree finite.

Example: simultaneous bugs detection (Abstract Diagnosis)

+ **Program:**

$$R_1\colon double(0) \to s(0) \qquad R_2\colon double(s(x)) \to s(double(x))$$

+ **Abstract Specification:** with $\kappa = 2$

$$\mathcal{S}^\kappa := \begin{cases} double(x) \mapsto \{\varepsilon \cdot \bot,\ \{x/0\} \cdot \dot{0},\ \{x/s(x')\} \cdot \dot{s}(s(\hat{y})), \\ \qquad \{x/s(0)\} \cdot \dot{s}(s(\dot{\hat{y}})),\ \{x/s(s(\hat{x}_1)\} \cdot \dot{s}(s(\dot{\hat{y}}))\} \end{cases}$$

We detect that both R_1 and R_2 are abstractly incorrect:

$$\mathcal{P}^\kappa[\![\{R_1\}]\!]_{\mathcal{S}^\kappa} = \{\varepsilon \cdot \bot,\ \{x/0\} \cdot \dot{s}(0)\} \not\leq \mathcal{S}^\kappa$$

$$\mathcal{P}^\kappa[\![\{R_2\}]\!]_{\mathcal{S}^\kappa} = \{\varepsilon \cdot \bot,\ \varepsilon \cdot \dot{s}(\bot),\ \{x/s(0)\} \cdot \dot{s}(\dot{0}),\ \{x/s(s(\hat{x}_1)\} \cdot \dot{s}(\dot{s}(\hat{y})),\ \{x/s(s(\hat{x}_1)\} \cdot \dot{s}(\dot{s}(\dot{\hat{y}}))\} \not\leq \mathcal{S}^\kappa$$

in fact it suffices $\kappa = 1$

Example: symptoms interference (Abstract Diagnosis)

Consider the buggy program

$$main = C(h(f(x)), x) \qquad h(s(x)) = 0 \qquad R\colon f(s(x)) = s(0)$$

where rule R should have been $f(x) = s(h(x))$ to be correct w.r.t. the intended semantics on $depth(k)$, with $k > 2$,

$$\mathcal{S}^{\kappa} = \begin{cases} f(x) \mapsto \{\varepsilon \cdot \bot,\ \varepsilon \cdot \dot{s}(\bot),\ \{x/s(x')\} \cdot \dot{s}(\dot{0})\} \\ h(x) \mapsto \{\varepsilon \cdot \bot,\ \{x/s(x')\} \cdot \dot{0}\} \\ main \mapsto \{\varepsilon \cdot \bot,\ \varepsilon \cdot \dot{C}(\bot, x),\ \varepsilon \cdot \dot{C}(\dot{0}, x)\} \end{cases}$$

The bug preserves the computed answer behavior both for h and f, but not for *main*. In fact, *main* evaluates to $\{x/s(x')\} \cdot C(0, s(x'))$. Rule R is abstractly incorrect:

$$\mathcal{P}^{\kappa}[\![\{R\}]\!]_{\mathcal{S}^{\kappa}} = \left\{ f(x) \mapsto \{\varepsilon \cdot \bot,\ \{x/s(x')\} \cdot \dot{s}(0)\} \right\} \not\leq \mathcal{S}^{\kappa}$$

Conclusion

+ Summary

- + Fix-point characterization:
  - ✔ correctly models the typical features of FL languages
  - ✘ does not handle H.O. and Residuation
  - ✔ goal-indipendent & compositional
- + Abstract Diagnosis:
  - ✔ consists in a static test
  - ✔ does not need any symptom in advice
  - ✔ points out more then one bug
  - ✔ can check if the specified property $\mathcal{S}^\alpha$ holds in $\mathcal{R}$
  - ✘ can give warnings even if there is no bug (false positives)
  - ✘ does not detect bugs not affecting the observed property

+ Future work

- + applying A.D. technique for more interesting properties
- + extend our results to Higher-Order and Residuation