Calculational Design of Semantics of the Eager Lambda-Calculus by Abstract Interpretation

Patrick Cousot

Joint work with

Radhia Cousot

WG 2.3 — Cambridge meeting — Cambridge, UK — July 25, 2008

Contents

1. Motivation and Objective

Motivation

– Static analysis requires the definition of the semantics of programming languages (i.e. models of runtime computations of programs) at various levels of abstraction:
  - finite — erroneous — infinite computations
  - traces — sets of states — input/output relations
  - small-step — big-step

Objective

– We look for a formalism to specify abstract semantics

– Handling uniformly the many different styles of presentations found in the literature (rules, fixpoints, equations, constraints, ...)

– A *non-monotone* generalization of inductive definitions from sets to posets seems adequate

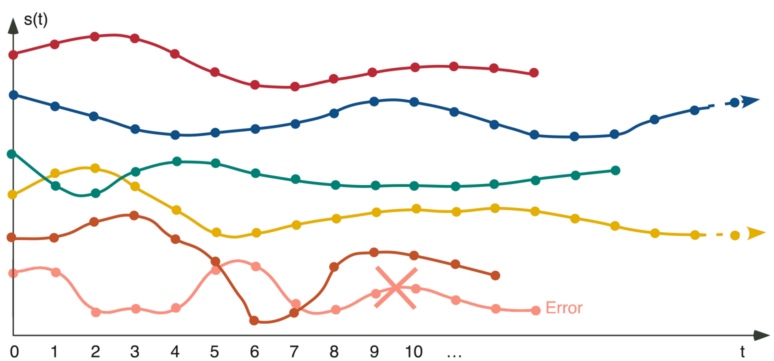
– Illustrated on the eager λ-calculus

2. Abstraction

Reference

[1] P. Cousot. Méthodes itératives de construction et d'approximation de points fixes d'opérateurs monotones sur un treillis, analyse sémantique de programmes. Thèse ès sciences mathématiques, University of Grenoble, March 1978.

Bifinitary Trace Semantics

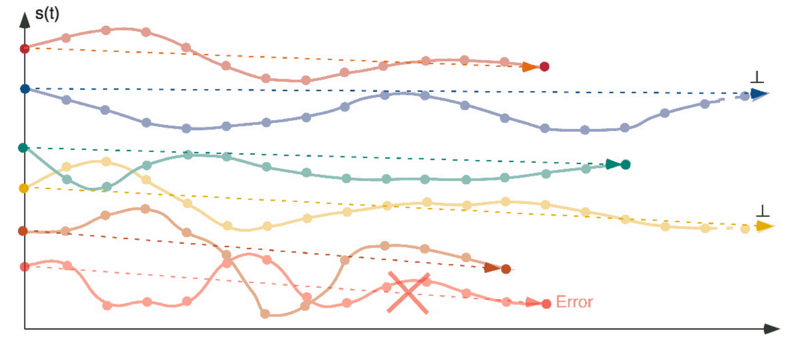


Traces

– $\mathbb{T}$ of states (e.g. terms)
– $\mathbb{T}^+$, set of nonempty finite sequences of states
– $\mathbb{T}^\omega$, set of infinite sequences of states
– $\mathbb{T}^\infty \triangleq \mathbb{T}^+ \cup \mathbb{T}^\omega$, nonempty finite or infinite sequences
– ϵ is the empty sequence $\epsilon \bullet \sigma = \sigma \bullet \epsilon = \sigma$
– $|\sigma| \in \mathbb{N} \cup \{\omega\}$ is the length of σ with $|\epsilon| = 0$
– If $\sigma \in \mathbb{T}^+$ then $|\sigma| > 0$ and $\sigma = \sigma_0 \bullet \sigma_1 \bullet \ldots \bullet \sigma_{|\sigma|-1}$
– If $\sigma \in \mathbb{T}^\omega$ then $|\sigma| = \omega$ and $\sigma = \sigma_0 \bullet \ldots \bullet \sigma_n \bullet \ldots$

Trace to Bifinitary Relational Semantics Abstraction

Bifinitary Relational Semantics = α(Trace Semantics)

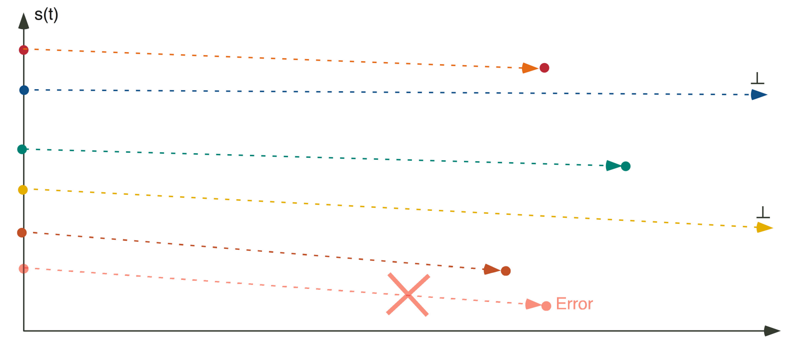


Abstraction to the Bifinitary Relational Semantics

remember the input/output behaviors,
forget about the intermediate computation steps

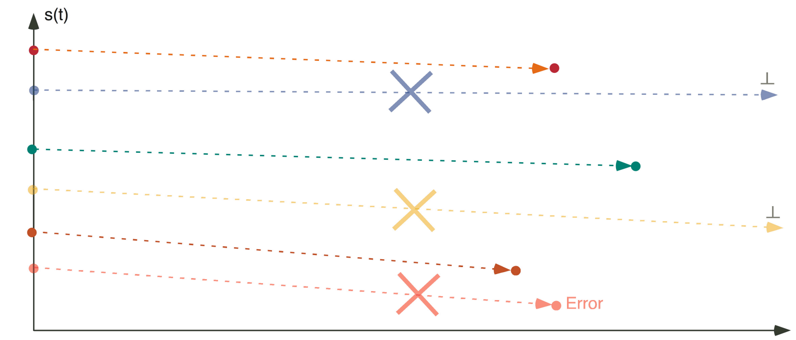
$$\alpha(T) \triangleq \{\alpha(\sigma) \mid \sigma \in T\}$$
$$\alpha(\sigma_0 \bullet \sigma_1 \bullet \ldots \bullet \sigma_n) \triangleq \sigma_0 \Longrightarrow \sigma_n$$
$$\alpha(\sigma_0 \bullet \ldots \bullet \sigma_n \bullet \ldots) \triangleq \sigma_0 \Longrightarrow \bot$$

Bifinitary Relational Semantics



Bifinitary to Finitary Relational Semantics Abstraction

Finitary Relational Semantics $= \alpha$(Relational Semantics)



Abstraction to the Finitary Relational Semantics

remember the finite input/output behaviors,
forget about non-termination

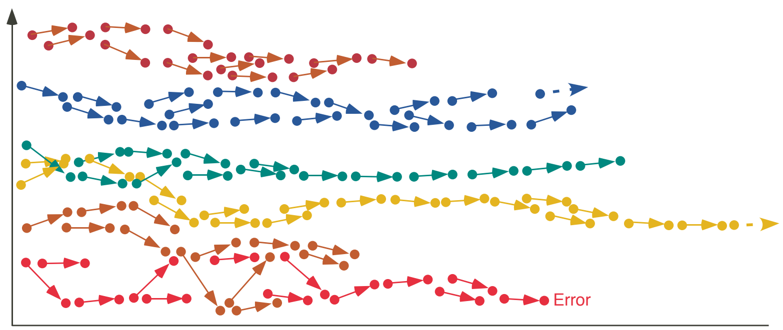
$$\alpha(T) \triangleq \bigcup\{\alpha(\sigma) \mid \sigma \in T\}$$

$$\alpha(\sigma_0 \Longrightarrow \sigma_n) \triangleq \{\sigma_0 \Longrightarrow \sigma_n\}$$

$$\alpha(\sigma_0 \Longrightarrow \bot) \triangleq \varnothing$$

Trace to Small-Step Operational Semantics Abstraction

Transition Semantics = α(Trace Semantics)



Abstraction to the Transition Semantics

remember execution steps,
forget about their sequencing

$$\alpha(T) \triangleq \bigcup\{\alpha(\sigma) \mid \sigma \in T\}$$

$$\alpha(\sigma_0 \bullet \sigma_1 \bullet \ldots \bullet \sigma_n) \triangleq \{\sigma_i \longrightarrow \sigma_{i+1} \mid 0 \leqslant i < n\}$$

$$\alpha(\sigma_0 \bullet \ldots \bullet \sigma_n \bullet \ldots) \triangleq \{\sigma_i \longrightarrow \sigma_{i+1} \mid i \geqslant 0\}$$

3. Bi-inductive Structural Definitions

Over-simplified for the presentation!

Inductive definitions

Set-theoretic [Acz77]

$\langle \wp(\mathcal{U}), \subseteq \rangle$	universe
$\frac{P}{c} \in \mathcal{R} \quad (P \in \wp(\mathcal{U}), c \in \mathcal{U})$	rules
$F(X) \triangleq \left\{c \;\middle|\; \exists \frac{P}{c} \in \mathcal{R} : P \subseteq X\right\}$	transformer
$\mathsf{lfp}^{\subseteq} F \in \wp(\mathcal{U})$	fixpoint def.

Inductive definitions

Set-theoretic [Acz77]	Order-theoretic [CC92]	
$\langle \wp(\mathcal{U}),\ \subseteq \rangle$	$\langle \mathcal{D},\ \sqsubseteq \rangle$	universe
$\frac{P}{c} \in \mathcal{R} \quad (P \in \wp(\mathcal{U}), c \in \mathcal{U})$	$\frac{P}{C} \in \mathcal{R} \quad (P, C \in \mathcal{D})$	rules
$F(X) \triangleq \left\{c \mid \exists \frac{P}{c} \in \mathcal{R} : P \subseteq X\right\}$	$F(X) \triangleq \bigsqcup \left\{C \mid \exists \frac{P}{C} \in \mathcal{R} : P \sqsubseteq X\right\}$	transformer
$\mathsf{lfp}^{\subseteq}\ F \in \wp(\mathcal{U})$	$\mathsf{lfp}^{\sqsubseteq}\ F \in \mathcal{D}$	fixpoint def.

Inductive definitions

Set-theoretic [Acz77]	Order-theoretic [CC92]	
$\langle \wp(\mathcal{U}),\ \subseteq \rangle$	$\langle \mathcal{D},\ \sqsubseteq \rangle$	universe
$\frac{P}{c} \in \mathcal{R} \quad (P \in \wp(\mathcal{U}), c \in \mathcal{U})$	$\frac{P}{C} \in \mathcal{R} \quad (P, C \in \mathcal{D})$	rules
$F(X) \triangleq \left\{c \mid \exists \frac{P}{c} \in \mathcal{R} : P \subseteq X\right\}$	$F(X) \triangleq \bigsqcup \left\{C \mid \exists \frac{P}{C} \in \mathcal{R} : P \sqsubseteq X\right\}$	transformer
$\mathsf{lfp}^{\subseteq}\ F \in \wp(\mathcal{U})$	$\mathsf{lfp}^{\sqsubseteq}\ F \in \mathcal{D}$	fixpoint def.

Existence of F ($\bigsqcup$) and $\mathsf{lfp}^{\sqsubseteq}\ F$?

4. Semantics of the Eager/Call by value λ-calculus

Syntax

Syntax of the Eager λ-calculus

$x, y, z, \ldots \in \mathbb{X}$		variables
$c \in \mathbb{C}$		constants $(\mathbb{X} \cap \mathbb{C} = \varnothing)$
$c ::= 0 \mid 1 \mid \ \ldots$		
$f \in \mathbb{F}$		function values
$f ::= \boldsymbol{\lambda} x \cdot a$		
$v \in \mathbb{V}$		values
$v ::= c \mid f$		
$e \in \mathbb{E}$		errors
$e ::= c\ a \mid e\ a \mid a\ e$		
$a, a', a_1, \ldots, b, , \ldots \in \mathbb{T}$		terms
$a ::= x \mid v \mid a\ a'$		

Small-Step Operational Semantics

Transition Semantics of the Eager λ-calculus [Plo81]

$$((\boldsymbol{\lambda} x \cdot a)\ v) \longrightarrow a[x \leftarrow v]^{1}, \quad v \in \mathbb{V}$$

$$\frac{a_0 \longrightarrow a_1}{a_0\ b \longrightarrow a_1\ b} \subseteq$$

$$\frac{b_0 \longrightarrow b_1}{f\ b_0 \longrightarrow f\ b_1} \subseteq, \quad f \in \mathbb{F}\ .$$

[1] Note: $a[x \leftarrow b]$ is the capture-avoiding substitution of b for all free occurences of x within a. We let FV(a) be the free variables of a. We define the call-by-value semantics of closed terms (without free variables) $\overline{\mathbb{T}} \triangleq \{a \in \mathbb{T} \mid \mathrm{FV}(a) = \varnothing\}$.

Example I: Finite Computation

function argument

$((\boldsymbol{\lambda} x \cdot x\ x)\ (\boldsymbol{\lambda} y \cdot y))\ ((\boldsymbol{\lambda} z \cdot z)\ 0)$

$\longrightarrow$ evaluate function

$((\boldsymbol{\lambda} y \cdot y)\ (\boldsymbol{\lambda} y \cdot y))\ ((\boldsymbol{\lambda} z \cdot z)\ 0)$

$\longrightarrow$ evaluate function, cont'd

$(\boldsymbol{\lambda} y \cdot y)\ ((\boldsymbol{\lambda} z \cdot z)\ 0)$

$\longrightarrow$ evaluate argument

$(\boldsymbol{\lambda} y \cdot y)\ 0$

$\longrightarrow$ apply function to argument

0 *a value!*

Example II: Infinite Computation

function argument

$(\boldsymbol{\lambda}\, x \cdot x\; x)\; (\boldsymbol{\lambda}\, x \cdot x\; x)$

$\longrightarrow$ apply function to argument

$(\boldsymbol{\lambda}\, x \cdot x\; x)\; (\boldsymbol{\lambda}\, x \cdot x\; x)$

$\longrightarrow$ apply function to argument

$(\boldsymbol{\lambda}\, x \cdot x\; x)\; (\boldsymbol{\lambda}\, x \cdot x\; x)$

$\longrightarrow$ apply function to argument

... *non-termination!*

Example III: Erroneous Computation

function argument

$((\boldsymbol{\lambda}\, x \cdot x\; x)\; ((\boldsymbol{\lambda}\, z \cdot z)\; 0))$

$\longrightarrow$ evaluate argument

$((\boldsymbol{\lambda}\, x \cdot x\; x)\; 0)$

$\longrightarrow$ apply function to argument

$(0\; 0)$

a runtime error!

Fixpoint Transition Semantics of the Eager λ-calculus

$$\begin{aligned}\Phi(X) \triangleq\; & \{((\boldsymbol{\lambda}\, x \cdot a)\; v) \longrightarrow a[x \leftarrow v] \mid v \in \mathbb{V}\}\\ & \cup \{a_0\; b \longrightarrow a_1\; b \mid a_0 \longrightarrow a_1 \in X\}\\ & \cup \{f\; b_0 \longrightarrow f\; b_1 \mid f \in \mathbb{F} \wedge b_0 \longrightarrow b_1 \in X\}\;.\end{aligned}$$

- Φ is $\subseteq$-monotonic on the complete lattice $\langle \wp(\mathbb{T} \times \mathbb{T}), \subseteq\rangle$
- So the transition semantics $\mathsf{lfp}^{\subseteq}\, \Phi$ is well-defined.

Finitary Relational Semantics

Finitary Relational Semantics

- Finite behaviors
- No infinite behavior
- No erroneous behavior
- Relation: term $\Longrightarrow$ result
- Can be presented in small-step [Plo81] or big-step [Kah88] style

Small-Step Finitary Semantics of the Eager λ-calculus

$$v \Longrightarrow v, \quad v \in \mathbb{V}$$

$$\frac{b \Longrightarrow v}{a \Longrightarrow v} \subseteq, \quad a \longrightarrow b$$

- $f(X) \triangleq \{v \Longrightarrow v \mid v \in \mathbb{V}\} \cup \{a \Longrightarrow v \mid b \Longrightarrow v \in X \wedge a \longrightarrow b\}$ is $\subseteq$-monotonic on the complete lattice $\langle \wp(\mathbb{T} \times \mathbb{V}), \subseteq \rangle$
- so $\mathsf{lfp}^{\subseteq} f$ does exist

Big-Step Finitary Semantics of the Eager λ-calculus

$$v \Longrightarrow v, \quad v \in \mathbb{V}$$

$$\frac{a[x \leftarrow v] \Longrightarrow r}{(\lambda\, x \cdot a)\ v \Longrightarrow r} \subseteq, \quad v, r \in \mathbb{V}$$

$$\frac{b \Longrightarrow v, \quad f\ v \Longrightarrow r}{f\ b \Longrightarrow r} \subseteq, \quad f, v, r \in \mathbb{V}$$

$$\frac{a \Longrightarrow f, \quad f\ b \Longrightarrow r}{a\ b \Longrightarrow r} \subseteq, \quad f, r \in \mathbb{V}\ .$$

Big-Step Finitary Semantics of the Eager λ-calculus

$$v \Longrightarrow v, \quad v \in \mathbb{V}$$

$$\frac{a[x \leftarrow v] \Longrightarrow r}{(\lambda\, x \cdot a)\ v \Longrightarrow r} \subseteq, \quad v, r \in \mathbb{V}$$

$$\frac{b \Longrightarrow v, \quad f\ v \Longrightarrow r}{f\ b \Longrightarrow r} \subseteq, \quad f, v, r \in \mathbb{V}$$

$$\frac{a \Longrightarrow f, \quad f\ b \Longrightarrow r}{a\ b \Longrightarrow r} \subseteq, \quad f, r \in \mathbb{V}\ .$$

Big-Step Finitary Semantics of the Eager λ-calculus

$$\mathsf{v} \Longrightarrow \mathsf{v}, \quad \mathsf{v} \in \mathbb{V}$$

$$\frac{\mathsf{a}[\mathsf{x} \leftarrow \mathsf{v}] \Longrightarrow r}{(\boldsymbol{\lambda}\,\mathsf{x} \cdot \mathsf{a})\ \mathsf{v} \Longrightarrow r} \subseteq, \quad \mathsf{v}, r \in \mathbb{V}$$

$$\frac{\mathsf{b} \Longrightarrow \mathsf{v}, \quad \mathsf{f}\ \mathsf{v} \Longrightarrow r}{\mathsf{f}\ \mathsf{b} \Longrightarrow r} \subseteq, \quad \mathsf{f}, \mathsf{v}, r \in \mathbb{V}$$

$$\frac{\mathsf{a} \Longrightarrow \mathsf{f}, \quad \mathsf{f}\ \mathsf{b} \Longrightarrow r}{\mathsf{a}\ \mathsf{b} \Longrightarrow r} \subseteq, \quad \mathsf{f}, r \in \mathbb{V}\ .$$

Big-Step Finitary Semantics of the Eager λ-calculus

$$\mathsf{v} \Longrightarrow \mathsf{v}, \quad \mathsf{v} \in \mathbb{V}$$

$$\frac{\mathsf{a}[\mathsf{x} \leftarrow \mathsf{v}] \Longrightarrow r}{(\boldsymbol{\lambda}\,\mathsf{x} \cdot \mathsf{a})\ \mathsf{v} \Longrightarrow r} \subseteq, \quad \mathsf{v}, r \in \mathbb{V}$$

$$\frac{\mathsf{b} \Longrightarrow \mathsf{v}, \quad \mathsf{f}\ \mathsf{v} \Longrightarrow r}{\mathsf{f}\ \mathsf{b} \Longrightarrow r} \subseteq, \quad \mathsf{f}, \mathsf{v}, r \in \mathbb{V}$$

$$\frac{\mathsf{a} \Longrightarrow \mathsf{f}, \quad \mathsf{f}\ \mathsf{b} \Longrightarrow r}{\mathsf{a}\ \mathsf{b} \Longrightarrow r} \subseteq, \quad \mathsf{f}, r \in \mathbb{V}\ .$$

Letf-to-right: the function is evaluated before the value parameter.

Big-Step Finitary Semantics of the Eager λ-calculus

$$\begin{aligned} F(X) \triangleq\ & \{\mathsf{v} \Longrightarrow \mathsf{v} \mid \mathsf{v} \in \mathbb{V}\} \\ & \cup \{(\boldsymbol{\lambda}\,\mathsf{x} \cdot \mathsf{a})\ \mathsf{v} \Longrightarrow r \mid \mathsf{a}[\mathsf{x} \leftarrow \mathsf{v}] \Longrightarrow r \wedge \mathsf{v}, r \in \mathbb{V}\} \\ & \cup \{\mathsf{f}\ \mathsf{b} \Longrightarrow r \mid \mathsf{b} \Longrightarrow \mathsf{v} \wedge \mathsf{f}\ \mathsf{v} \Longrightarrow r \wedge \mathsf{f}, r, \mathsf{v} \in \mathbb{V}\} \\ & \cup \{\mathsf{a}\ \mathsf{b} \Longrightarrow r \mid \mathsf{a} \Longrightarrow \mathsf{f} \wedge \mathsf{f}\ \mathsf{b} \Longrightarrow r \wedge \mathsf{f}, r \in \mathbb{V}\} \end{aligned}$$

- F is $\subseteq$-monotonic on the complete lattice $\langle \wp(\mathbb{T} \times \mathbb{V}), \subseteq \rangle$
- so $\mathsf{lfp}^{\subseteq} F$ does exist.

Adding divergence: Bifinitary relational semantics

Bifinitary Relational Semantics

- Finite behaviors
- Infinite behaviors
- No erroneous behavior
- Relation: term $\Longrightarrow$ result or term $\Longrightarrow \bot$
- Can be presented in small-step or big-step style

The Computational Ordering [CC92]

- The semantic domain $\wp(\mathbb{T} \times (\mathbb{V} \cup \{\bot\}))$ is partitionned into finite $\wp(\mathbb{T} \times \mathbb{V})$ and infinite $\wp(\mathbb{T} \times \{\bot\})$ behaviors
- $X^+ \triangleq X \cap (\mathbb{T} \times \mathbb{V})$ finite behaviors in X
- $X^\omega \triangleq X \cap (\mathbb{T} \times \{\bot\})$ infinite behaviors in X
- $X \sqsubseteq Y \triangleq (X^+ \subseteq Y^+) \wedge (X^\omega \supseteq Y^\omega)$ computational ordering [2]
- $\langle \wp(\mathbb{T} \times (\mathbb{V} \cup \{\bot\})), \sqsubseteq \rangle$ is a complete lattice [3]

[2] more finite behaviors and less infinite behaviors, so induction for finite behaviors and co-induction for infinite behaviors

[3] with lub $\bigsqcup_{i\in\Delta} X_i \triangleq \bigcup_{i\in\Delta} X_i^+ \cup \bigcap_{i\in\Delta} X_i^\omega$

Small-Step Bifinitary Relational Semantics of the Eager λ-Calculus

$$\mathsf{v} \Longrightarrow \mathsf{v}, \quad \mathsf{v} \in \mathbb{V}$$

$$\frac{\mathsf{b} \Longrightarrow r}{\mathsf{a} \Longrightarrow r}\sqsubseteq, \quad \mathsf{a} \longrightarrow \mathsf{b}, \quad r \in \mathbb{V} \cup \{\bot\}$$

- $f(X) \triangleq \{\mathsf{v} \Longrightarrow \mathsf{v} \mid \mathsf{v} \in \mathbb{V}\} \cup \{\mathsf{a} \Longrightarrow \mathsf{v} \mid \mathsf{b} \Longrightarrow \mathsf{v} \in X \wedge \mathsf{a} \longrightarrow \mathsf{b}\}$ is $\sqsubseteq$-monotonic on the complete lattice $\langle \wp(\mathbb{T} \times (\mathbb{V} \cup \{\bot\})), \sqsubseteq \rangle$
- so $\mathsf{lfp}^{\sqsubseteq} f$ does exist

Reference

[2] P. Cousot. Constructive Design of a Hierarchy of Semantics of a Transition System by Abstract Interpretation. *Theoretical Computer Science* 277(1–2):47–103, 2002.

Big-Step Bifinitary Relational Semantics of the Eager λ-calculus

$$\mathsf{v} \Longrightarrow \mathsf{v}, \quad \mathsf{v} \in \mathbb{V}$$

$$\frac{\mathsf{a} \Longrightarrow \bot}{\mathsf{a}\ \mathsf{b} \Longrightarrow \bot}\sqsubseteq \qquad \frac{\mathsf{b} \Longrightarrow \bot}{\mathsf{f}\ \mathsf{b} \Longrightarrow \bot}\sqsubseteq, \quad \mathsf{f} \in \mathbb{V}$$

$$\frac{\mathsf{a}[\mathsf{x} \leftarrow \mathsf{v}] \Longrightarrow r}{(\boldsymbol{\lambda}\,\mathsf{x}\cdot\mathsf{a})\ \mathsf{v} \Longrightarrow r}\sqsubseteq, \quad \mathsf{v} \in \mathbb{V},\ r \in \mathbb{V} \cup \{\bot\}$$

$$\frac{\mathsf{b} \Longrightarrow \mathsf{v}, \quad \mathsf{f}\ \mathsf{v} \Longrightarrow r}{\mathsf{f}\ \mathsf{b} \Longrightarrow r}\sqsubseteq, \quad \mathsf{f}, \mathsf{v} \in \mathbb{V},\ r \in \mathbb{V} \cup \{\bot\}$$

$$\frac{\mathsf{a} \Longrightarrow \mathsf{f}, \quad \mathsf{f}\ \mathsf{b} \Longrightarrow r}{\mathsf{a}\ \mathsf{b} \Longrightarrow r}\sqsubseteq, \quad \mathsf{f} \in \mathbb{V},\ r \in \mathbb{V} \cup \{\bot\}\ .$$

Fixpoint Big-Step Bifinitary Semantics of the Eager λ-calculus

$$
\begin{array}{rl}
F(X) \triangleq & \{\mathsf{v} \Longrightarrow \mathsf{v} \mid \mathsf{v} \in \mathbb{V}\} \\
& \cup\ \{\mathsf{a}\ \mathsf{b} \Longrightarrow \bot \mid \mathsf{a} \Longrightarrow \bot \vee \mathsf{b} \Longrightarrow \bot\} \\
& \cup\ \{(\boldsymbol{\lambda}\,\mathsf{x} \cdot \mathsf{a})\ \mathsf{v} \Longrightarrow r \mid \mathsf{a}[\mathsf{x} \leftarrow \mathsf{v}] \Longrightarrow r \wedge \\
& \qquad \mathsf{v} \in \mathbb{V} \wedge r \in \mathbb{V} \cup \{\bot\}\} \\
& \cup\ \{\mathsf{f}\ \mathsf{b} \Longrightarrow r \mid \mathsf{b} \Longrightarrow \mathsf{v} \wedge \mathsf{f}\ \mathsf{v} \Longrightarrow \mathsf{f} \wedge \\
& \qquad \mathsf{v} \in \mathbb{V} \wedge r \in \mathbb{V} \cup \{\bot\}\} \\
& \cup\ \{\mathsf{a}\ \mathsf{b} \Longrightarrow r \mid \mathsf{a} \Longrightarrow \mathsf{f} \wedge \mathsf{f}\ \mathsf{b} \Longrightarrow r \wedge \\
& \qquad \mathsf{f} \in \mathbb{V} \wedge r \in \mathbb{V} \cup \{\bot\}\}
\end{array}
$$

Which Order for Which Fixpoint?

- F is $\subseteq$-monotonic on $\langle \wp(\mathbb{T} \times (\mathbb{V} \cup \{\bot\})),\ \subseteq\rangle$.
- However the definition is problematic, because:
  - $\mathsf{lfp}^{\subseteq} F$ exists, but induction yields only finite behaviors!
  - $\mathsf{gfp}^{\subseteq} F$ exists, but co-induction yields spurious finite behaviors!
  - F is not monotonic for the computational ordering $\sqsubseteq$, so *the existence of* $\mathsf{lfp}^{\sqsubseteq} F$ *is questionable!*

Induction Yields Only Finite Behaviors!

- $F^0 = \varnothing$ contains only finite behaviors
- by induction hypothesis F^δ hence $F^{\delta+1} \triangleq F(F^\delta)$ contain only finite behaviors
- by induction hypothesis F^δ, $\delta < \lambda$ hence $F^\lambda \triangleq \bigcup_{\delta<\lambda} F^\delta$ contain only finite behaviors
- so $\mathsf{lfp}^{\subseteq} F = F^\epsilon$ contains only finite behaviors!

Co-Induction Yields Spurious Finite Behaviors!

- For $\theta \triangleq \boldsymbol{\lambda}\,\mathsf{x} \cdot (\mathsf{x}\ \mathsf{x})$, $(\mathsf{x}\ \mathsf{x})[\mathsf{x} \leftarrow \theta] = \theta\ \theta$ so $(\theta\ \theta) \longrightarrow (\theta\ \theta)$
- $F^0 = \mathbb{T} \times (\mathbb{V} \cup \{\bot\})$ contains the behavior $(\theta\ \theta) \Longrightarrow 0$
- if, by co-induction hypothesis, $(\theta\ \theta) \Longrightarrow 0 \in F^\delta$ then $F^{\delta+1} \triangleq F(F^\delta)$ contains $(\theta\ \theta) \Longrightarrow 0$ by $\dfrac{\mathsf{a}[\mathsf{x} \leftarrow \mathsf{v}] \Longrightarrow r}{(\boldsymbol{\lambda}\,\mathsf{x} \cdot \mathsf{a})\ \mathsf{v} \Longrightarrow r} \supseteq$
- if, by co-induction hypothesis, $(\theta\ \theta) \Longrightarrow 0 \in F^\delta$, $\delta < \lambda$ then $F^\lambda \triangleq \bigcap_{\delta<\lambda} F^\delta$ contains $(\theta\ \theta) \Longrightarrow 0$
- so $\mathsf{gfp}^{\subseteq} F = F^\epsilon$ contains $(\theta\ \theta) \Longrightarrow 0$!

This is a spurious finite behavior since $(\theta\ \theta)$ always diverges: $(\theta\ \theta) \Longrightarrow \bot$.

Non-monotonicity for the Computational Ordering $\sqsubseteq$

F is not $\sqsubseteq$-monotonic on the complete lattice $\langle \wp(\mathbb{T} \times (\mathbb{V} \cup \{\bot\})),\ \sqsubseteq \rangle$

- Let $\theta \triangleq \boldsymbol{\lambda}\, \mathsf{x} \cdot (\mathsf{x}\ \mathsf{x})$ such that $(\theta\ \theta) \Longrightarrow \bot$
- $X \triangleq \{(\theta\ \theta) \Longrightarrow \bot\}$
- $Y \triangleq \{(\boldsymbol{\lambda}\, \mathsf{x} \cdot \mathsf{x}\ \theta) \Longrightarrow \theta,\ (\theta\ \theta) \Longrightarrow \bot\}$
- $X \sqsubseteq Y$
- $((\boldsymbol{\lambda}\, \mathsf{x} \cdot \mathsf{x}\ \theta)\ \theta) \Longrightarrow \bot \in F(Y)$ by $\dfrac{(\boldsymbol{\lambda}\, \mathsf{x} \cdot \mathsf{x}\ \theta) \Longrightarrow \theta, \quad \theta\ \theta \Longrightarrow \bot}{(\boldsymbol{\lambda}\, \mathsf{x} \cdot \mathsf{x}\ \theta)\ \theta \Longrightarrow \bot}\sqsubseteq$
- $((\boldsymbol{\lambda}\, \mathsf{x} \cdot \mathsf{x}\ \theta)\ \theta) \Longrightarrow \bot \notin F(X)$
- so $F(X) \not\sqsubseteq F(Y)$

Classical fixpoint theorems are inapplicable.

Existence of $\mathsf{lfp}^{\sqsubseteq} F$?

- $\mathsf{lfp}^{\subseteq}\ \boldsymbol{\lambda} X \cdot (F(X^{+}))^{+}$ is the set of finite computations
- $\mathsf{gfp}^{\subseteq}\ \boldsymbol{\lambda} Y \cdot (F(X^{+} \cup Y^{\omega}))^{\omega}$ is the set of infinite computations built out of given finite computations in X^{+}
- The set of finite and infinite computations is

$$\mathsf{lfp}^{\subseteq}\ \boldsymbol{\lambda} X \cdot (F(X^{+}))^{+} \cup \mathsf{gfp}^{\subseteq}\ \boldsymbol{\lambda} Y \cdot (F(\mathsf{lfp}^{\subseteq}\ \boldsymbol{\lambda} X \cdot (F(X^{+}))^{+} \cup Y^{\omega}))^{\omega}$$

$$= \mathsf{lfp}^{\sqsubseteq}\ F$$

- so $\mathsf{lfp}^{\sqsubseteq} F$ does exist

Adequacy of the Small-Step $\mathsf{lfp}^{\sqsubseteq} f$ and Big-Step $\mathsf{lfp}^{\sqsubseteq} F$ Bifinitary Relational Semantics

- The small-step $\mathsf{lfp}^{\sqsubseteq} f$ and big-step $\mathsf{lfp}^{\sqsubseteq} F$ bifinitary relational semantics are the abstraction of corresponding small-step $\mathsf{lfp}^{\sqsubseteq} \vec{f}$ and big-step $\mathsf{lfp}^{\sqsubseteq} \vec{F}$ bifinitary trace semantics
- Both small-step $\mathsf{lfp}^{\sqsubseteq} \vec{f}$ and big-step $\mathsf{lfp}^{\sqsubseteq} \vec{F}$ trace semantics coincide with the traces generated by the transitional semantics

Bifinitary Trace Semantics

The Computational Ordering for Traces

Given $X, Y \in \wp(\mathbb{T}^\infty)$, we define

- $X^+ \triangleq X \cap \mathbb{T}^+$ finite traces
- $X^\omega \triangleq X \cap \mathbb{T}^\omega$ infinite traces
- $X \sqsubseteq Y \triangleq X^+ \subseteq Y^+ \wedge X^\omega \supseteq Y^\omega$ computational order
- $\langle \wp(\mathbb{T}^\infty), \sqsubseteq, \mathbb{T}^\omega, \mathbb{T}^+, \sqcup, \sqcap \rangle$ is a complete lattice [3]

Reference

[3] P. Cousot and R. Cousot. Inductive Definitions, Semantics and Abstract Interpretation. In *Conference Record of the 19th ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Programming Languages*, pages 83–94, Albuquerque, New Mexico, 1992. ACM Press, New York, U.S.A.

Small-Step Bifinitary Trace Semantics

Small-Step Bifinitary Trace Semantics

$$\mathsf{v}, \quad \mathsf{v} \in \mathbb{V}$$

$$\frac{\mathsf{b} \bullet \sigma}{\mathsf{a} \bullet \mathsf{b} \bullet \sigma}\sqsubseteq, \quad \mathsf{a} \longrightarrow \mathsf{b}$$

- $\vec{f}(X) \triangleq \{\mathsf{v} \mid \mathsf{v} \in \mathbb{V}\} \cup \{\mathsf{a} \bullet \mathsf{b} \bullet \sigma \mid \mathsf{a} \longrightarrow \mathsf{b} \wedge \mathsf{b} \bullet \sigma \in X\}$
- $\vec{f}$ is $\sqsubseteq$-monotonic on the complete lattice $\langle \wp(\mathbb{T}^\infty), \sqsubseteq \rangle$
- $\mathsf{lfp}^{\sqsubseteq} \vec{f}$ does exist

Reference

[4] P. Cousot. Constructive Design of a Hierarchy of Semantics of a Transition System by Abstract Interpretation. *Theoretical Computer Science* 277(1–2):47–103, 2002.

Big-Step Bifinitary Trace Semantics

Operations on Traces

- For $a \in \mathbb{T}$ and $\sigma \in \mathbb{T}^\infty$, we define $a@\sigma$ to be $\sigma' \in \mathbb{T}^\infty$ such that $\forall i < |\sigma| : \sigma'_i = a\ \sigma_i$
- The application $a@\sigma$ of term a to trace σ is

$$\sigma \;=\; \sigma_0 \bullet\!\!-\; \sigma_1 \bullet\!\!-\; \sigma_2 \bullet\!\!-\; \sigma_3 \bullet\!\!-\; \ldots\; \sigma_i \bullet\!\!-\; \ldots$$

$$a@\sigma \;=\; a\ \sigma_0 \bullet\!\!-\; a\ \sigma_1 \bullet\!\!-\; a\ \sigma_2 \bullet\!\!-\; a\ \sigma_3 \bullet\!\!-\; \ldots\; a\ \sigma_i \bullet\!\!-\; \ldots$$

Operations on Traces (Cont'd)

- Similarly for $a \in \mathbb{T}$ and $\sigma \in \mathbb{T}^\infty$, $\sigma@a$ is σ' where $\forall i < |\sigma| : \sigma'_i = \sigma_i\ a$
- The application $\sigma@a$ trace σ to term a is

$$\sigma \;=\; \sigma_0 \bullet\!\!-\; \sigma_1 \bullet\!\!-\; \sigma_2 \bullet\!\!-\; \sigma_3 \bullet\!\!-\; \ldots\; \sigma_i \bullet\!\!-\; \ldots$$

$$\sigma@a \;=\; \sigma_0\ a \bullet\!\!-\; \sigma_1\ a \bullet\!\!-\; \sigma_2\ a \bullet\!\!-\; \sigma_3\ a \bullet\!\!-\; \ldots\; \sigma_i\ a \bullet\!\!-\; \ldots$$

Big-Step Bifinitary Trace Semantics $\vec{\mathbb{S}}$ of the Eager λ-calculus

$$\mathsf{v} \in \vec{\mathbb{S}},\ \mathsf{v} \in \mathbb{V} \qquad \frac{\mathsf{a}[\mathsf{x} \leftarrow \mathsf{v}] \bullet \sigma \in \vec{\mathbb{S}}}{(\boldsymbol{\lambda}\,\mathsf{x} \cdot \mathsf{a})\ \mathsf{v} \bullet \mathsf{a}[\mathsf{x} \leftarrow \mathsf{v}] \bullet \sigma \in \vec{\mathbb{S}}}\sqsubseteq,\ \mathsf{v} \in \mathbb{V}$$

$$\frac{\sigma \in \vec{\mathbb{S}}^\omega}{\mathsf{f}@\sigma \in \vec{\mathbb{S}}}\sqsubseteq,\ \mathsf{f} \in \mathbb{V} \qquad \frac{\sigma \bullet \mathsf{v} \in \vec{\mathbb{S}}^+,\ (\mathsf{f}\ \mathsf{v}) \bullet \sigma' \in \vec{\mathbb{S}}}{(\mathsf{f}@\sigma) \bullet (\mathsf{f}\ \mathsf{v}) \bullet \sigma' \in \vec{\mathbb{S}}}\sqsubseteq,\ \mathsf{f}, \mathsf{v} \in \mathbb{V}$$

$$\frac{\sigma \in \vec{\mathbb{S}}^\omega}{\sigma@\mathsf{b} \in \vec{\mathbb{S}}}\sqsubseteq \qquad \frac{\sigma \bullet \mathsf{f} \in \vec{\mathbb{S}}^+,\ (\mathsf{f}\ \mathsf{b}) \bullet \sigma' \in \vec{\mathbb{S}}}{(\sigma@\mathsf{b}) \bullet (\mathsf{f}\ \mathsf{b}) \bullet \sigma' \in \vec{\mathbb{S}}}\sqsubseteq,\ \mathsf{f} \in \mathbb{V}$$

Big-Step Bifinitary Trace Semantics $\vec{\mathbb{S}}$ of the Eager λ-calculus

$$\mathsf{v} \in \vec{\mathbb{S}},\ \mathsf{v} \in \mathbb{V} \qquad \frac{\mathsf{a}[\mathsf{x} \leftarrow \mathsf{v}] \bullet \sigma \in \vec{\mathbb{S}}}{(\boldsymbol{\lambda}\,\mathsf{x} \cdot \mathsf{a})\ \mathsf{v} \bullet \mathsf{a}[\mathsf{x} \leftarrow \mathsf{v}] \bullet \sigma \in \vec{\mathbb{S}}}\sqsubseteq,\ \mathsf{v} \in \mathbb{V}$$

$$\frac{\sigma \in \vec{\mathbb{S}}^\omega}{\mathsf{f}@\sigma \in \vec{\mathbb{S}}}\sqsubseteq,\ \mathsf{f} \in \mathbb{V} \qquad \frac{\sigma \bullet \mathsf{v} \in \vec{\mathbb{S}}^+,\ (\mathsf{f}\ \mathsf{v}) \bullet \sigma' \in \vec{\mathbb{S}}}{(\mathsf{f}@\sigma) \bullet (\mathsf{f}\ \mathsf{v}) \bullet \sigma' \in \vec{\mathbb{S}}}\sqsubseteq,\ \mathsf{f}, \mathsf{v} \in \mathbb{V}$$

$$\frac{\sigma \in \vec{\mathbb{S}}^\omega}{\sigma@\mathsf{b} \in \vec{\mathbb{S}}}\sqsubseteq \qquad \frac{\sigma \bullet \mathsf{f} \in \vec{\mathbb{S}}^+,\ (\mathsf{f}\ \mathsf{b}) \bullet \sigma' \in \vec{\mathbb{S}}}{(\sigma@\mathsf{b}) \bullet (\mathsf{f}\ \mathsf{b}) \bullet \sigma' \in \vec{\mathbb{S}}}\sqsubseteq,\ \mathsf{f} \in \mathbb{V}$$

WG 2.3, Cambridge, 7/25/2008 — 56 — P. Cousot

Big-Step Bifinitary Trace Semantics $\vec{\mathbb{S}}$ of the Eager λ-calculus

$$v \in \vec{\mathbb{S}},\ v \in \mathbb{V} \qquad \frac{a[x \leftarrow v] \bullet \sigma \in \vec{\mathbb{S}}}{(\boldsymbol{\lambda}\, x \cdot a)\ v \bullet a[x \leftarrow v] \bullet \sigma \in \vec{\mathbb{S}}} \sqsubseteq,\ v \in \mathbb{V}$$

$$\frac{\sigma \in \vec{\mathbb{S}}^{\omega}}{f@\sigma \in \vec{\mathbb{S}}} \sqsubseteq,\ f \in \mathbb{V} \qquad \frac{\sigma \bullet v \in \vec{\mathbb{S}}^{+},\ (f\ v) \bullet \sigma' \in \vec{\mathbb{S}}}{(f@\sigma) \bullet (f\ v) \bullet \sigma' \in \vec{\mathbb{S}}} \sqsubseteq,\ f, v \in \mathbb{V}$$

$$\frac{\sigma \in \vec{\mathbb{S}}^{\omega}}{\sigma@b \in \vec{\mathbb{S}}} \sqsubseteq \qquad \frac{\sigma \bullet f \in \vec{\mathbb{S}}^{+},\ (f\ b) \bullet \sigma' \in \vec{\mathbb{S}}}{(\sigma@b) \bullet (f\ b) \bullet \sigma' \in \vec{\mathbb{S}}} \sqsubseteq,\ f \in \mathbb{V}$$

Big-Step Bifinitary Trace Semantics $\vec{\mathbb{S}}$ of the Eager λ-calculus

$$v \in \vec{\mathbb{S}},\ v \in \mathbb{V} \qquad \frac{a[x \leftarrow v] \bullet \sigma \in \vec{\mathbb{S}}}{(\boldsymbol{\lambda}\, x \cdot a)\ v \bullet a[x \leftarrow v] \bullet \sigma \in \vec{\mathbb{S}}} \sqsubseteq,\ v \in \mathbb{V}$$

$$\frac{\sigma \in \vec{\mathbb{S}}^{\omega}}{f@\sigma \in \vec{\mathbb{S}}} \sqsubseteq,\ f \in \mathbb{V} \qquad \frac{\sigma \bullet v \in \vec{\mathbb{S}}^{+},\ (f\ v) \bullet \sigma' \in \vec{\mathbb{S}}}{(f@\sigma) \bullet (f\ v) \bullet \sigma' \in \vec{\mathbb{S}}} \sqsubseteq,\ f, v \in \mathbb{V}$$

$$\frac{\sigma \in \vec{\mathbb{S}}^{\omega}}{\sigma@b \in \vec{\mathbb{S}}} \sqsubseteq \qquad \frac{\sigma \bullet f \in \vec{\mathbb{S}}^{+},\ (f\ b) \bullet \sigma' \in \vec{\mathbb{S}}}{(\sigma@b) \bullet (f\ b) \bullet \sigma' \in \vec{\mathbb{S}}} \sqsubseteq,\ f \in \mathbb{V}$$

WG 2.3, Cambridge, 7/25/2008 — 56 — P. Cousot

Big-Step Bifinitary Trace Semantics $\vec{\mathbb{S}}$ of the Eager λ-calculus

$$v \in \vec{\mathbb{S}},\ v \in \mathbb{V} \qquad \frac{a[x \leftarrow v] \bullet \sigma \in \vec{\mathbb{S}}}{(\boldsymbol{\lambda}\, x \cdot a)\ v \bullet a[x \leftarrow v] \bullet \sigma \in \vec{\mathbb{S}}} \sqsubseteq,\ v \in \mathbb{V}$$

$$\frac{\sigma \in \vec{\mathbb{S}}^{\omega}}{f@\sigma \in \vec{\mathbb{S}}} \sqsubseteq,\ f \in \mathbb{V} \qquad \frac{\sigma \bullet v \in \vec{\mathbb{S}}^{+},\ (f\ v) \bullet \sigma' \in \vec{\mathbb{S}}}{(f@\sigma) \bullet (f\ v) \bullet \sigma' \in \vec{\mathbb{S}}} \sqsubseteq,\ f, v \in \mathbb{V}$$

$$\frac{\sigma \in \vec{\mathbb{S}}^{\omega}}{\sigma@b \in \vec{\mathbb{S}}} \sqsubseteq \qquad \frac{\sigma \bullet f \in \vec{\mathbb{S}}^{+},\ (f\ b) \bullet \sigma' \in \vec{\mathbb{S}}}{(\sigma@b) \bullet (f\ b) \bullet \sigma' \in \vec{\mathbb{S}}} \sqsubseteq,\ f \in \mathbb{V}$$

WG 2.3, Cambridge, 7/25/2008 — 56 — P. Cousot

Big-Step Bifinitary Trace Semantics $\vec{\mathbb{S}}$ of the Eager λ-calculus

$$v \in \vec{\mathbb{S}},\ v \in \mathbb{V} \qquad \frac{a[x \leftarrow v] \bullet \sigma \in \vec{\mathbb{S}}}{(\boldsymbol{\lambda}\, x \cdot a)\ v \bullet a[x \leftarrow v] \bullet \sigma \in \vec{\mathbb{S}}} \sqsubseteq,\ v \in \mathbb{V}$$

$$\frac{\sigma \in \vec{\mathbb{S}}^{\omega}}{f@\sigma \in \vec{\mathbb{S}}} \sqsubseteq,\ f \in \mathbb{V} \qquad \frac{\sigma \bullet v \in \vec{\mathbb{S}}^{+},\ (f\ v) \bullet \sigma' \in \vec{\mathbb{S}}}{(f@\sigma) \bullet (f\ v) \bullet \sigma' \in \vec{\mathbb{S}}} \sqsubseteq,\ f, v \in \mathbb{V}$$

$$\frac{\sigma \in \vec{\mathbb{S}}^{\omega}}{\sigma@b \in \vec{\mathbb{S}}} \sqsubseteq \qquad \frac{\sigma \bullet f \in \vec{\mathbb{S}}^{+},\ (f\ b) \bullet \sigma' \in \vec{\mathbb{S}}}{(\sigma@b) \bullet (f\ b) \bullet \sigma' \in \vec{\mathbb{S}}} \sqsubseteq,\ f \in \mathbb{V}$$

WG 2.3, Cambridge, 7/25/2008 — 56 — P. Cousot

Fixpoint Big-Step Bifinitary Trace Semantics

$$\begin{aligned}\vec{F}(X) \triangleq\ & \{v \in \overline{\mathbb{T}}^{\infty} \mid v \in \mathbb{V}\} \cup \\ & \{(\lambda x \cdot a)\ v \bullet a[x \leftarrow v] \bullet \sigma \mid v \in \mathbb{V} \wedge a[x \leftarrow v] \bullet \sigma \in X\} \cup \\ & \{\sigma @ b \mid \sigma \in X^{\omega}\} \cup \\ & \{(\sigma @ b) \bullet (f\ b) \bullet \sigma' \mid \sigma \neq \epsilon \wedge \sigma \bullet f \in X^{+} \wedge f \in \mathbb{V} \wedge \\ & \qquad (f\ b) \bullet \sigma' \in X\} \cup \\ & \{f @ \sigma \mid f \in \mathbb{V} \wedge \sigma \in X^{\omega}\} \cup \\ & \{(f @ \sigma) \bullet (f\ v) \bullet \sigma' \mid f, v \in \mathbb{V} \wedge \sigma \neq \epsilon \wedge \sigma \bullet v \in X^{+} \wedge \\ & \qquad (f\ v) \bullet \sigma' \in X\}\ .\end{aligned}$$

$\vec{F}$ is $\subseteq$-monotonic on $\wp(\overline{\mathbb{T}}^{\infty})$.

Existence of the Fixpoint $\mathsf{lfp}^{\sqsubseteq}\ \vec{F}$

- $\mathsf{lfp}^{\subseteq}\ \vec{F}$ (finite traces) and $\mathsf{gfp}^{\subseteq}\ \vec{F}$ (spurious finite traces) are inadequate
- $\vec{F}$ is not $\sqsubseteq$-monotonic
- Nevertheless $\mathsf{lfp}^{\sqsubseteq}\ \vec{F}$ does exist
- So the big-step bifinitary trace semantics can be well-defined as

$$\mathsf{lfp}^{\sqsubseteq}\ \vec{F}$$

Characterization of the Small-Step & Big-Step Bifinitary Trace Semantics

Characterization of the Fixpoint Small-Step and Big-Step Bifinitary Trace Semantics

- $\mathsf{lfp}^{\sqsubseteq}\ \vec{f}$ collects the finite and infinite traces generated by the transitional semantics [5]

$$\begin{aligned}\mathsf{lfp}^{\sqsubseteq}\ \vec{f} = \quad & \{\sigma_0 \bullet \sigma_1 \bullet \ldots \bullet \sigma_n \in \mathbb{T}^{+} \mid \forall i \in [0, n-1] : \sigma_i \longrightarrow \sigma_{i+1} \\ & \qquad \wedge\ \sigma_n \in \mathbb{V}\} \\ & \cup\ \{\sigma_0 \bullet \sigma_1 \bullet \ldots \bullet \sigma_i \bullet \ldots \in \mathbb{T}^{\omega} \mid \forall i \geqslant 0 : \sigma_i \longrightarrow \sigma_{i+1}\}\end{aligned}$$

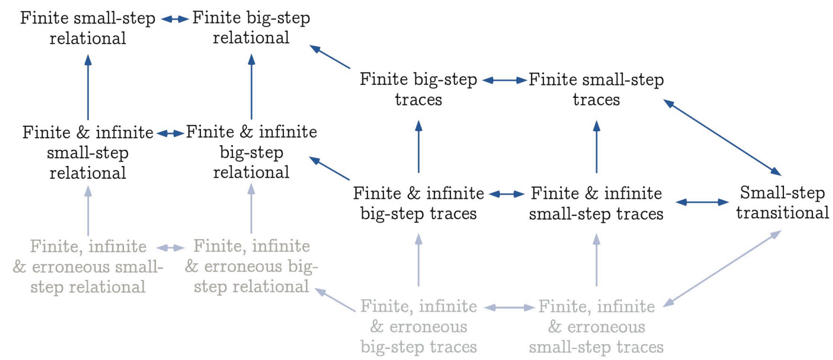
- $\mathsf{lfp}^{\sqsubseteq}\ \vec{f} = \mathsf{lfp}^{\sqsubseteq}\ \vec{F}$

Reference

[5] P. Cousot. Constructive Design of a Hierarchy of Semantics of a Transition System by Abstract Interpretation. *Theoretical Computer Science* 277(1–2):47–103, 2002.

5. Conclusion

The Hierarchy of Semantics for the Eager λ-Calculus



Conclusion

– In proofs [CC85, CC87] and static analysis (e.g. strictness, [Myc80], typing [Cou97, Ler06]), both finite and infinite behaviors have to be taken into account

– Such proof methods and static analyzes must be proved correct with respect to a semantics chosen at various levels of abstraction (small-step/big-step – finitary/bifinitary – relational/trace)

– Static analyzes use various equivalent presentations (fixpoints, equational, constraints and inference rules)

– The SOS bifinitary extension should satisfy these needs.

The End

6. Bibliography

[Acz77] P. Aczel. An introduction to inductive definitions. In J. Barwise, editor, *Handbook of Mathematical Logic*, volume 90 of *Studies in Logic and the Foundations of Mathematics*, pages 739–782. Elsevier Science Publishers B.V., Amsterdam, Pays-Bas, 1977.

[CC85] P. Cousot and R. Cousot. 'À la Floyd' induction principles for proving inevitability properties of programs, chapitre invité. In M. Nivat and J. Reynolds, editors, *Algebraic Methods in Semantics*, chapter 8, pages 277–312. Cambridge University Press, Cambridge, Royaume Uni, 1985.

[CC87] P. Cousot and R. Cousot. Sometime = always + recursion ≡ always: on the equivalence of the intermittent and invariant assertions methods for proving inevitability properties of programs. *Acta Informatica*, 24:1–31, 1987.

[CC92] P. Cousot and R. Cousot. Inductive definitions, semantics and abstract interpretation. In *Conference Record of the Ninthteenth Annual ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages*, pages 83–94, Albuquerque, Nouveau Mexique, USA, 1992. ACM Press, New York, New York, USA.

[Cou97] P. Cousot. Types as abstract interpretations, papier invité. In *Conference Record of the Twentyfourth Annual ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages*, pages 316–331, Paris, janvier 1997. ACM Press, New York, New York, USA.

[Kah88] G. Kahn. Natural semantics. In K. Fuchi and M. Nivat, editors, *Programming of Future Generation Computers*, pages 237–258. Elsevier Science Publishers B.V., Amsterdam, Pays-Bas, 1988.

[Ler06] X. Leroy. Coinductive big-step operational semantics. In P. Sestoft, editor, *Proceedings of the Fifteenth European Symposium on Programming Languages and Systems, ESOP '2006*, Vienne, Autriche, Lecture Notes in Computer Science 3924, pages 54–68. Springer, Berlin, Allemagne, 27–28 mars 2006.

[Myc80] A. Mycroft. The theory and practice of transforming call-by-need into call-by-value. In B. Robinet, editor, *Proceedings of the Fourth International Symposium on Programming*, Paris, 22–24 avril 1980, Lecture Notes in Computer Science 83, pages 270–281. Springer, Berlin, Allemagne, 1980.

[Plo81] G.D. Plotkin. A structural approach to operational semantics. Technical Report DAIMI FN-19, Aarhus University, Danemark, septembre 1981.