

Principles of Program Analysis:

Control Flow Analysis

Transparencies based on Chapter 3 of the book: Flemming Nielson, Hanne Riis Nielson and Chris Hankin: [Principles of Program Analysis](#). Springer Verlag 2005. ©Flemming Nielson & Hanne Riis Nielson & Chris Hankin.

The Dynamic Dispatch Problem

<pre> : [call p(p1,1,v)]ℓ_c^1 : [call p(p2,2,v)]ℓ_c^2 : </pre>		<pre> proc p(procval q, val x, res y) isℓ_n : [call q (x,y)]ℓ_c^p : endℓ_x </pre>	<p>which procedure is called?</p>
--	--	---	---------------------------------------

These problems arise for:

- imperative languages with procedures as parameters
- object oriented languages
- functional languages

Example:

```
let f = fn x => x 1;  
    g = fn y => y+2;  
    h = fn z => z+3  
in (f g) + (f h)
```

The aim of **Control Flow Analysis**:

For each function application, which functions may be applied?

Control Flow Analysis computes the interprocedural flow relation used when formulating interprocedural Data Flow Analysis.

Syntax of the Fun Language

Syntactic categories:

$e \in$	Exp	expressions (or labelled terms)
$t \in$	Term	terms (or unlabelled expressions)
$f, x \in$	Var	variables
$c \in$	Const	constants
$op \in$	Op	binary operators
$\ell \in$	Lab	labels

Syntax:

$e ::= t^\ell$

$t ::= c \mid x \mid \text{fn } x \Rightarrow e_0 \mid \text{fun } f \ x \Rightarrow e_0 \mid e_1 \ e_2$
 $\mid \text{if } e_0 \text{ then } e_1 \text{ else } e_2 \mid \text{let } x = e_1 \text{ in } e_2 \mid e_1 \ op \ e_2$

(Labels correspond to program points or nodes in the parse tree.)

Examples:

- $((\text{fn } x \Rightarrow x^1)^2 (\text{fn } y \Rightarrow y^3)^4)^5$
- $(\text{let } f = (\text{fn } x \Rightarrow (x^1 \ 1^2)^3)^4;$
 $\text{in } (\text{let } g = (\text{fn } y \Rightarrow y^5)^6;$
 $\text{in } (\text{let } h = (\text{fn } z \Rightarrow z^7)^8$
 $\text{in } ((f^9 \ g^{10})^{11} + (f^{12} \ h^{13})^{14})^{15})^{16})^{17})^{18}$
- $(\text{let } g = (\text{fun } f \ x \Rightarrow (f^1 (\text{fn } y \Rightarrow y^2)^3)^4)^5$
 $\text{in } (g^6 (\text{fn } z \Rightarrow z^7)^8)^9)^{10}$

Abstract 0-CFA Analysis

- Abstract domains
- Specification of the analysis
- Well-definedness of the analysis

Towards defining the Abstract Domains

The *result* of a 0-CFA analysis is a pair $(\hat{\mathcal{C}}, \hat{\rho})$:

- $\hat{\mathcal{C}}$ is the *abstract cache* associating abstract values with each labelled program point
- $\hat{\rho}$ is the *abstract environment* associating abstract values with each variable

Example:

$((\text{fn } x \Rightarrow x^1)^2 (\text{fn } y \Rightarrow y^3)^4)^5$

Three guesses of a 0-CFA analysis result:

	$(\hat{C}_e, \hat{\rho}_e)$	$(\hat{C}'_e, \hat{\rho}'_e)$	$(\hat{C}''_e, \hat{\rho}''_e)$
1	$\{\text{fn } y \Rightarrow y^3\}$	$\{\text{fn } y \Rightarrow y^3\}$	$\{\text{fn } x \Rightarrow x^1, \text{fn } y \Rightarrow y^3\}$
2	$\{\text{fn } x \Rightarrow x^1\}$	$\{\text{fn } x \Rightarrow x^1\}$	$\{\text{fn } x \Rightarrow x^1, \text{fn } y \Rightarrow y^3\}$
3	\emptyset	\emptyset	$\{\text{fn } x \Rightarrow x^1, \text{fn } y \Rightarrow y^3\}$
4	$\{\text{fn } y \Rightarrow y^3\}$	$\{\text{fn } y \Rightarrow y^3\}$	$\{\text{fn } x \Rightarrow x^1, \text{fn } y \Rightarrow y^3\}$
5	$\{\text{fn } y \Rightarrow y^3\}$	$\{\text{fn } y \Rightarrow y^3\}$	$\{\text{fn } x \Rightarrow x^1, \text{fn } y \Rightarrow y^3\}$
x	$\{\text{fn } y \Rightarrow y^3\}$	\emptyset	$\{\text{fn } x \Rightarrow x^1, \text{fn } y \Rightarrow y^3\}$
y	\emptyset	\emptyset	$\{\text{fn } x \Rightarrow x^1, \text{fn } y \Rightarrow y^3\}$

Example:

```
(let g = (fun f x => (f1 (fn y => y2)3)4)5
  in (g6 (fn z => z7)8)9)10
```

Abbreviations:

$$\begin{aligned} f &= \text{fun } f \ x \Rightarrow (f^1 \ (\text{fn } y \Rightarrow y^2)^3)^4 \\ \text{id}_y &= \text{fn } y \Rightarrow y^2 \\ \text{id}_z &= \text{fn } z \Rightarrow z^7 \end{aligned}$$

One guess of a 0-CFA analysis result:

$\hat{C}_{lp}(1) = \{f\}$	$\hat{C}_{lp}(6) = \{f\}$	$\hat{\rho}_{lp}(f) = \{f\}$
$\hat{C}_{lp}(2) = \emptyset$	$\hat{C}_{lp}(7) = \emptyset$	$\hat{\rho}_{lp}(g) = \{f\}$
$\hat{C}_{lp}(3) = \{\text{id}_y\}$	$\hat{C}_{lp}(8) = \{\text{id}_z\}$	$\hat{\rho}_{lp}(x) = \{\text{id}_y, \text{id}_z\}$
$\hat{C}_{lp}(4) = \emptyset$	$\hat{C}_{lp}(9) = \emptyset$	$\hat{\rho}_{lp}(y) = \emptyset$
$\hat{C}_{lp}(5) = \{f\}$	$\hat{C}_{lp}(10) = \emptyset$	$\hat{\rho}_{lp}(z) = \emptyset$

Abstract Domains

Formally:

$$\hat{v} \in \widehat{\mathbf{Val}} = \mathcal{P}(\mathbf{Term}) \quad \textit{abstract values}$$

$$\hat{\rho} \in \widehat{\mathbf{Env}} = \mathbf{Var} \rightarrow \widehat{\mathbf{Val}} \quad \textit{abstract environments}$$

$$\hat{C} \in \widehat{\mathbf{Cache}} = \mathbf{Lab} \rightarrow \widehat{\mathbf{Val}} \quad \textit{abstract caches}$$

An abstract value \hat{v} is a set of terms of the forms

- $\mathbf{fn} \ x \Rightarrow e_0$
- $\mathbf{fun} \ f \ x \Rightarrow e_0$

Control Flow Analysis versus Use-Definition chains

The aim: to trace how **definition points** reach **use points**

- Control Flow Analysis
 - **definition points**: where function abstractions are created
 - **use points**: where functions are applied
- Use-Definition chains
 - **definition points**: where variables are assigned a value
 - **use points**: where values of variables are accessed

Specification of the 0-CFA

When is a proposed guess $(\hat{C}, \hat{\rho})$ of an analysis results an *acceptable 0-CFA analysis* for the program?

Different approaches:

- abstract specification
- syntax-directed and constraint-based specifications
- algorithms for computing the *best* result

Specification of the Abstract 0-CFA

$(\hat{\mathbf{C}}, \hat{\rho}) \models e$ means that $(\hat{\mathbf{C}}, \hat{\rho})$ is an *acceptable Control Flow Analysis* of the expression e

The relation \models has functionality:

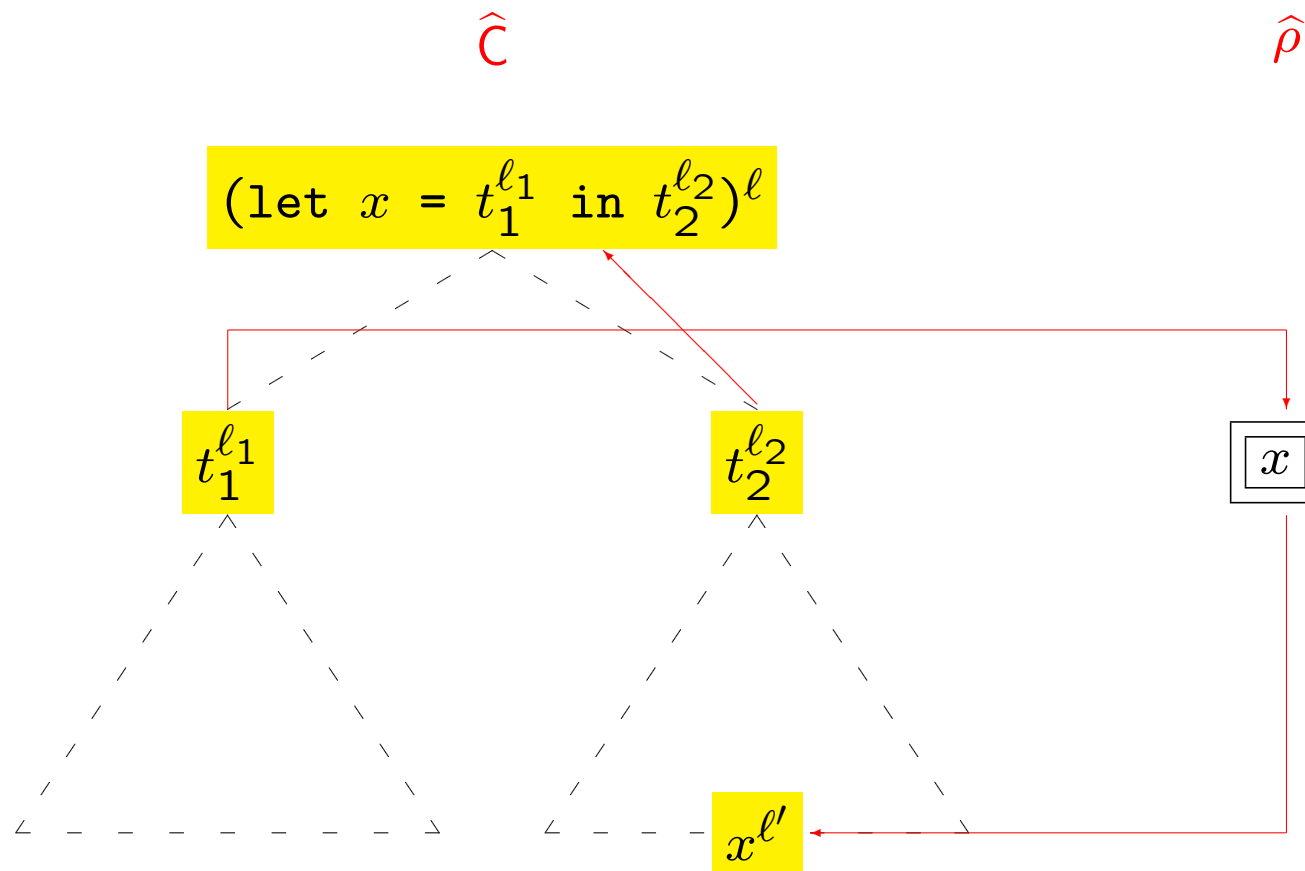
$$\models : (\widehat{\mathbf{Cache}} \times \widehat{\mathbf{Env}} \times \mathbf{Exp}) \rightarrow \{true, false\}$$

Clauses for Abstract 0-CFA (1)

$$(\hat{C}, \hat{\rho}) \models c^\ell \text{ always}$$

$$(\hat{C}, \hat{\rho}) \models x^\ell \quad \underline{\text{iff}} \quad \hat{\rho}(x) \subseteq \hat{C}(\ell)$$

$$\begin{aligned} (\hat{C}, \hat{\rho}) \models (\text{let } x = t_1^{\ell_1} \text{ in } t_2^{\ell_2})^\ell \\ \underline{\text{iff}} \quad (\hat{C}, \hat{\rho}) \models t_1^{\ell_1} \wedge (\hat{C}, \hat{\rho}) \models t_2^{\ell_2} \wedge \\ \hat{C}(\ell_1) \subseteq \hat{\rho}(x) \wedge \hat{C}(\ell_2) \subseteq \hat{C}(\ell) \end{aligned}$$



Clauses for Abstract 0-CFA (2)

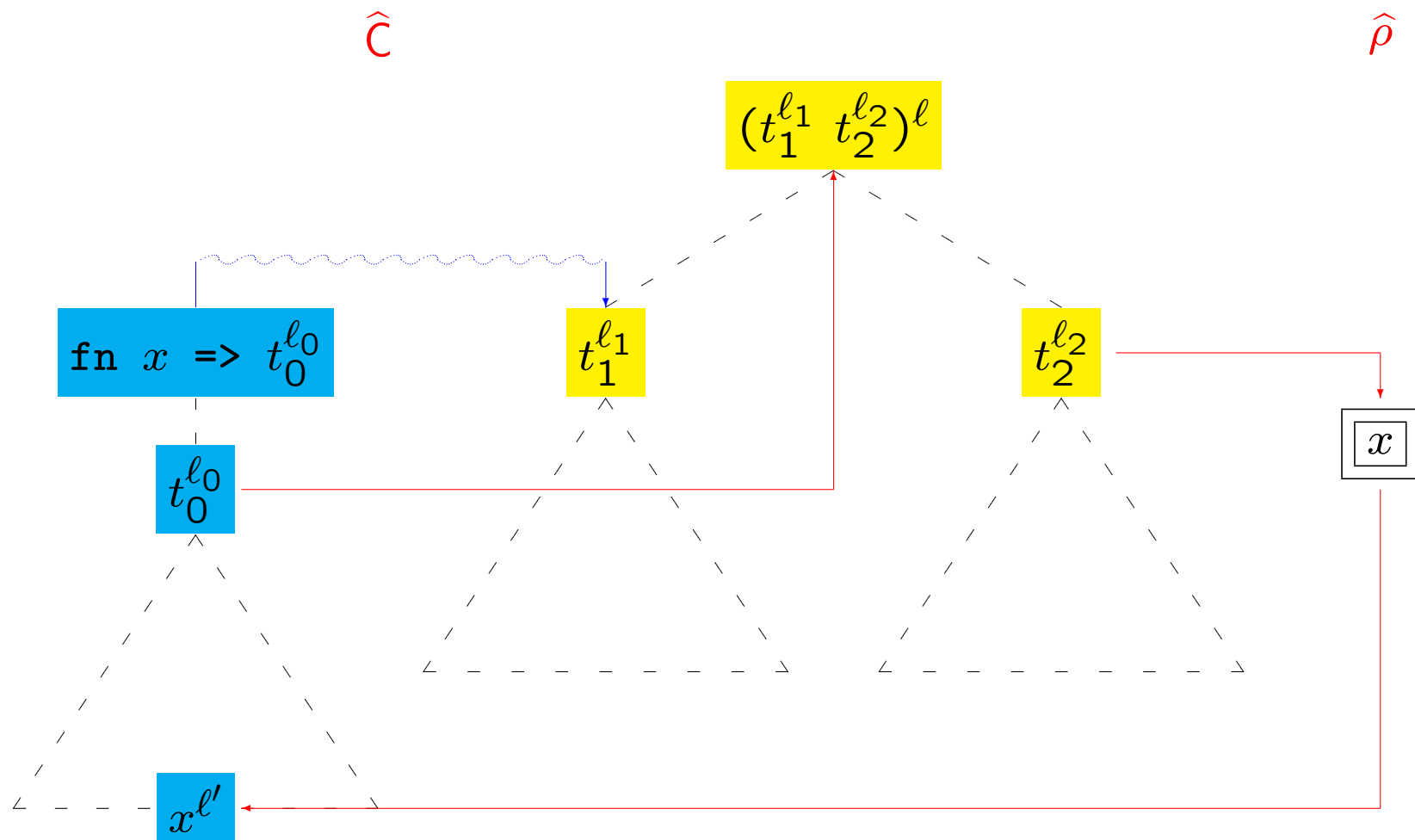
$$\begin{aligned} (\hat{\mathbf{C}}, \hat{\rho}) &\models (\text{if } t_0^{\ell_0} \text{ then } t_1^{\ell_1} \text{ else } t_2^{\ell_2})^\ell \\ \text{iff} \quad &(\hat{\mathbf{C}}, \hat{\rho}) \models t_0^{\ell_0} \wedge \\ &(\hat{\mathbf{C}}, \hat{\rho}) \models t_1^{\ell_1} \wedge (\hat{\mathbf{C}}, \hat{\rho}) \models t_2^{\ell_2} \wedge \\ &\hat{\mathbf{C}}(\ell_1) \subseteq \hat{\mathbf{C}}(\ell) \quad \wedge \quad \hat{\mathbf{C}}(\ell_2) \subseteq \hat{\mathbf{C}}(\ell) \end{aligned}$$

$$\begin{aligned} (\hat{\mathbf{C}}, \hat{\rho}) &\models (t_1^{\ell_1} \text{ op } t_2^{\ell_2})^\ell \\ \text{iff} \quad &(\hat{\mathbf{C}}, \hat{\rho}) \models t_1^{\ell_1} \wedge (\hat{\mathbf{C}}, \hat{\rho}) \models t_2^{\ell_2} \end{aligned}$$

Clauses for Abstract 0-CFA (3)

$$(\hat{C}, \hat{\rho}) \models (\text{fn } x \Rightarrow t_0^{\ell_0})^\ell \text{ iff } \{\text{fn } x \Rightarrow t_0^{\ell_0}\} \subseteq \hat{C}(\ell)$$

$$\begin{aligned} (\hat{C}, \hat{\rho}) \models (t_1^{\ell_1} \ t_2^{\ell_2})^\ell \\ \text{iff} \quad & (\hat{C}, \hat{\rho}) \models t_1^{\ell_1} \wedge (\hat{C}, \hat{\rho}) \models t_2^{\ell_2} \wedge \\ & (\forall (\text{fn } x \Rightarrow t_0^{\ell_0}) \in \hat{C}(\ell_1) : \quad (\hat{C}, \hat{\rho}) \models t_0^{\ell_0} \wedge \\ & \quad \hat{C}(\ell_2) \subseteq \hat{\rho}(x) \wedge \hat{C}(\ell_0) \subseteq \hat{C}(\ell)) \end{aligned}$$



Clauses for Abstract 0-CFA (4)

$$(\hat{C}, \hat{\rho}) \models (\text{fun } f \ x \Rightarrow e_0)^\ell \text{ iff } \{\text{fun } f \ x \Rightarrow e_0\} \subseteq \hat{C}(\ell)$$

$$(\hat{C}, \hat{\rho}) \models (t_1^{\ell_1} \ t_2^{\ell_2})^\ell$$

$$\text{iff } (\hat{C}, \hat{\rho}) \models t_1^{\ell_1} \ \wedge \ (\hat{C}, \hat{\rho}) \models t_2^{\ell_2} \ \wedge$$

$$(\forall (\text{fn } x \Rightarrow t_0^{\ell_0}) \in \hat{C}(\ell_1) : (\hat{C}, \hat{\rho}) \models t_0^{\ell_0} \ \wedge$$

$$\hat{C}(\ell_2) \subseteq \hat{\rho}(x) \ \wedge \ \hat{C}(\ell_0) \subseteq \hat{C}(\ell)) \ \wedge$$

$$(\forall (\text{fun } f \ x \Rightarrow t_0^{\ell_0}) \in \hat{C}(\ell_1) : (\hat{C}, \hat{\rho}) \models t_0^{\ell_0} \ \wedge$$

$$\hat{C}(\ell_2) \subseteq \hat{\rho}(x) \ \wedge \ \hat{C}(\ell_0) \subseteq \hat{C}(\ell) \ \wedge$$

$$\{\text{fun } f \ x \Rightarrow t_0^{\ell_0}\} \subseteq \hat{\rho}(f))$$

Example:

Two guesses for $((\text{fn } x \Rightarrow x^1)^2 (\text{fn } y \Rightarrow y^3)^4)^5$

	$(\hat{C}_e, \hat{\rho}_e)$	$(\hat{C}'_e, \hat{\rho}'_e)$
1	$\{\text{fn } y \Rightarrow y^3\}$	$\{\text{fn } y \Rightarrow y^3\}$
2	$\{\text{fn } x \Rightarrow x^1\}$	$\{\text{fn } x \Rightarrow x^1\}$
3	\emptyset	\emptyset
4	$\{\text{fn } y \Rightarrow y^3\}$	$\{\text{fn } y \Rightarrow y^3\}$
5	$\{\text{fn } y \Rightarrow y^3\}$	$\{\text{fn } y \Rightarrow y^3\}$
x	$\{\text{fn } y \Rightarrow y^3\}$	\emptyset
y	\emptyset	\emptyset

Checking the guesses:

$$(\hat{C}_e, \hat{\rho}_e) \models ((\text{fn } x \Rightarrow x^1)^2 (\text{fn } y \Rightarrow y^3)^4)^5$$

$$(\hat{C}'_e, \hat{\rho}'_e) \not\models ((\text{fn } x \Rightarrow x^1)^2 (\text{fn } y \Rightarrow y^3)^4)^5$$

Well-definedness of the Abstract 0-CFA

Difficulty: The clause for function application is *not* of a form that allows us to define $(\hat{C}, \hat{\rho}) \models e$ by Structural Induction in the expression e

$$(\hat{C}, \hat{\rho}) \models (t_1^{\ell_1} t_2^{\ell_2})^\ell$$

$$\text{iff } (\hat{C}, \hat{\rho}) \models t_1^{\ell_1} \wedge (\hat{C}, \hat{\rho}) \models t_2^{\ell_2} \wedge$$

$$(\forall (\text{fn } x \Rightarrow t_0^{\ell_0}) \in \hat{C}(\ell_1) : (\hat{C}, \hat{\rho}) \models t_0^{\ell_0} \wedge$$

$$\hat{C}(\ell_2) \subseteq \hat{\rho}(x) \wedge \hat{C}(\ell_0) \subseteq \hat{C}(\ell))$$

Solution: The relation \models is defined by **coinduction**, that is, as the **greatest fixed point** of a functional.

The functional \mathcal{Q}

The clauses for \models define a function:

$$\begin{aligned} \mathcal{Q} : ((\widehat{\text{Cache}} \times \widehat{\text{Env}} \times \text{Exp}) &\rightarrow \{true, false\}) \\ &\rightarrow ((\widehat{\text{Cache}} \times \widehat{\text{Env}} \times \text{Exp}) \rightarrow \{true, false\}) \end{aligned}$$

Example:

$$\begin{aligned} (\hat{\mathbf{C}}, \hat{\rho}) \models (\text{let } x = t_1^{\ell_1} \text{ in } t_2^{\ell_2})^\ell \\ \text{iff } (\hat{\mathbf{C}}, \hat{\rho}) \models t_1^{\ell_1} \wedge (\hat{\mathbf{C}}, \hat{\rho}) \models t_2^{\ell_2} \wedge \hat{\mathbf{C}}(\ell_1) \subseteq \hat{\rho}(x) \wedge \hat{\mathbf{C}}(\ell_2) \subseteq \hat{\mathbf{C}}(\ell) \end{aligned}$$

becomes

$$\begin{aligned} \mathcal{Q}(\mathcal{Q})(\hat{\mathbf{C}}, \hat{\rho}, (\text{let } x = t_1^{\ell_1} \text{ in } t_2^{\ell_2})^\ell) \\ = \mathcal{Q}(\hat{\mathbf{C}}, \hat{\rho}, t_1^{\ell_1}) \wedge \mathcal{Q}(\hat{\mathbf{C}}, \hat{\rho}, t_2^{\ell_2}) \wedge \hat{\mathbf{C}}(\ell_1) \subseteq \hat{\rho}(x) \wedge \hat{\mathbf{C}}(\ell_2) \subseteq \hat{\mathbf{C}}(\ell) \end{aligned}$$

Properties of \mathcal{Q}

\mathcal{Q} is a monotone function on the complete lattice

$$((\widehat{\mathbf{Cache}} \times \widehat{\mathbf{Env}} \times \mathbf{Exp}) \rightarrow \{true, false\}, \sqsubseteq)$$

where the ordering \sqsubseteq is defined by:

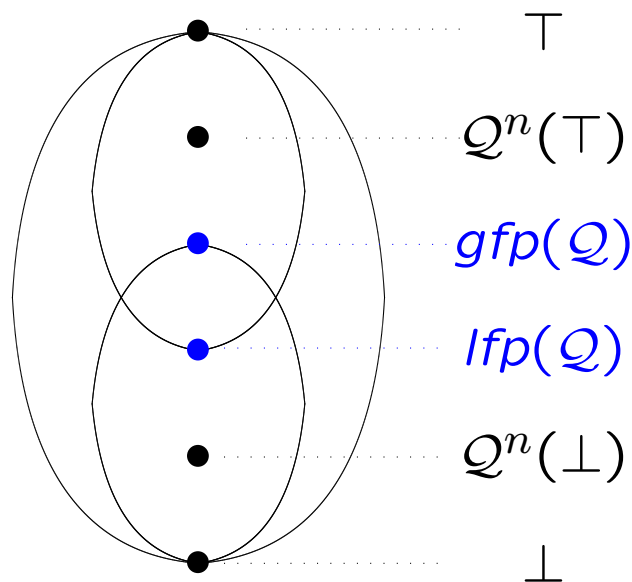
$$Q_1 \sqsubseteq Q_2 \text{ iff } \forall(\hat{\mathbf{C}}, \hat{\rho}, e) : (Q_1(\hat{\mathbf{C}}, \hat{\rho}, e) = true) \Rightarrow (Q_2(\hat{\mathbf{C}}, \hat{\rho}, e) = true)$$

Hence \mathcal{Q} has fixed points and we shall define \models coinductively:

\models is the *greatest fixed point* of \mathcal{Q}

Tarski's Theorem:

A monotone function on a complete lattice has a complete lattice of fixed points and in particular a least and a greatest fixed point.



$$Q : ((\widehat{\mathbf{Cache}} \times \widehat{\mathbf{Env}} \times \mathbf{Exp}) \rightarrow \{true, false\}) \\ \rightarrow ((\widehat{\mathbf{Cache}} \times \widehat{\mathbf{Env}} \times \mathbf{Exp}) \rightarrow \{true, false\})$$

Coinductive definition:

$$gfp(Q) = \bigsqcup \{P \mid Q(P) \sqsupseteq P\}$$

Inductive definition:

$$lfp(Q) = \bigsqcap \{P \mid Q(P) \sqsubseteq P\} \\ = \bigsqcup_n Q^n(\perp)$$

assuming that $Q(P)(\hat{\mathbf{C}}, \hat{\rho}, e)$ only depends on finitely many values of P

Inductive Definition

$$P = \text{Ifp}(\mathcal{Q}) = \bigsqcup_n \mathcal{Q}^n(\perp) \quad \text{assuming } \dots$$

P can be expressed as

$$P(\hat{C}, \hat{\rho}, x^\ell) \text{ iff } \hat{\rho}(x) \subseteq \hat{C}(\ell)$$

$$P(\hat{C}, \hat{\rho}, (\text{let } x = t_1^{\ell_1} \text{ in } t_2^{\ell_2})^\ell) \text{ iff} \\ P(\hat{C}, \hat{\rho}, t_1^{\ell_1}) \wedge P(\hat{C}, \hat{\rho}, t_2^{\ell_2}) \\ \hat{C}(\ell_1) \subseteq \hat{\rho}(x) \wedge \hat{C}(\ell_2) \subseteq \hat{C}(\ell)$$

⋮

simply because $P = \mathcal{Q}(P)$

Example:

0 is a number

$n+1$ is a number iff n is a number
(Peano's Axioms)

to check $P(\hat{C}, \hat{\rho}, e)$

simply unfold using the clauses:

if it terminates

and yields true: then it holds

and yields false: then it does not

if it loops

because it repeats itself:

then it does not hold

but we cannot detect it ...

Example:

$2 = 0+1+1$ is a number

because $0+1$ is because 0 is

Inductive Definition

to prove: $\forall(\hat{C}, \hat{\rho}, e) : P(\hat{C}, \hat{\rho}, e) \Rightarrow R(\hat{C}, \hat{\rho}, e)$

show: $R(\hat{C}, \hat{\rho}, x^\ell)$ if $\hat{\rho}(x) \subseteq \hat{C}(\ell)$ axiom

$$\frac{R(\hat{C}, \hat{\rho}, t_1^{\ell_1}) \quad R(\hat{C}, \hat{\rho}, t_2^{\ell_2})}{R(\hat{C}, \hat{\rho}, (\text{let } x = t_1^{\ell_1} \text{ in } t_2^{\ell_2})^\ell)}$$

inference rule

if $\hat{C}(\ell_1) \subseteq \hat{\rho}(x) \wedge \hat{C}(\ell_2) \subseteq \hat{C}(\ell)$

\vdots

Examples:

- mathematical induction: $R(0), \frac{R(n)}{R(n+1)}$
- structural induction
- induction on the shape of inference tree

Coinductive Definition

$$P = gfp(Q) = \bigsqcup \{R \mid R \sqsubseteq Q(R)\}$$

P can be expressed as

$$\begin{aligned} P(\hat{C}, \hat{\rho}, x^\ell) &\text{ iff } \hat{\rho}(x) \subseteq \hat{C}(\ell) \\ P(\hat{C}, \hat{\rho}, (\text{let } x = t_1^{\ell_1} \text{ in } t_2^{\ell_2})^\ell) &\text{ iff} \\ &P(\hat{C}, \hat{\rho}, t_1^{\ell_1}) \wedge P(\hat{C}, \hat{\rho}, t_2^{\ell_2}) \\ &\hat{C}(\ell_1) \subseteq \hat{\rho}(x) \wedge \hat{C}(\ell_2) \subseteq \hat{C}(\ell) \\ &\vdots \end{aligned}$$

simply because $P = Q(P)$

to check $P(\hat{C}, \hat{\rho}, e)$

find some R such that

$R(\hat{C}, \hat{\rho}, e)$ can be shown to hold

that is prove:

$$R(\hat{C}, \hat{\rho}, x^\ell) \text{ if } \hat{\rho}(x) \subseteq \hat{C}(\ell)$$

$$\frac{R(\hat{C}, \hat{\rho}, t_1^{\ell_1}) \quad R(\hat{C}, \hat{\rho}, t_2^{\ell_2})}{R(\hat{C}, \hat{\rho}, (\text{let } x = t_1^{\ell_1} \text{ in } t_2^{\ell_2})^\ell)}$$

$$\text{if } \hat{C}(\ell_1) \subseteq \hat{\rho}(x) \wedge \hat{C}(\ell_2) \subseteq \hat{C}(\ell)$$

$$\vdots$$

and use $P = \bigsqcup \{R \mid R \sqsubseteq Q(R)\}$

Coinductive Definition

to prove: $\forall(\hat{C}, \hat{\rho}, e) : P(\hat{C}, \hat{\rho}, e) \Rightarrow R(\hat{C}, \hat{\rho}, e)$

- try to prove it using $P = Q(P)$
i.e. by using the way P is expressed
- if it fails try to do induction (on the structure or size) of e
- if it fails ... you will need an extra insight

Example: loop

```
(let g = (fun f x => (f1 (fn y => y2)3)4)5
  in (g6 (fn z => z7)8)9)10
```

Abbreviations:

$$\begin{aligned} f &= \text{fun } f \ x \Rightarrow (f^1 \ (\text{fn } y \Rightarrow y^2)^3)^4 \\ \text{id}_y &= \text{fn } y \Rightarrow y^2 \\ \text{id}_z &= \text{fn } z \Rightarrow z^7 \end{aligned}$$

One guess of a 0-CFA analysis result:

$\hat{C}_{lp}(1) = \{f\}$	$\hat{C}_{lp}(6) = \{f\}$	$\hat{\rho}_{lp}(f) = \{f\}$
$\hat{C}_{lp}(2) = \emptyset$	$\hat{C}_{lp}(7) = \emptyset$	$\hat{\rho}_{lp}(g) = \{f\}$
$\hat{C}_{lp}(3) = \{\text{id}_y\}$	$\hat{C}_{lp}(8) = \{\text{id}_z\}$	$\hat{\rho}_{lp}(x) = \{\text{id}_y, \text{id}_z\}$
$\hat{C}_{lp}(4) = \emptyset$	$\hat{C}_{lp}(9) = \emptyset$	$\hat{\rho}_{lp}(y) = \emptyset$
$\hat{C}_{lp}(5) = \{f\}$	$\hat{C}_{lp}(10) = \emptyset$	$\hat{\rho}_{lp}(z) = \emptyset$

Naively checking the solution gives rise to circularity:

To show

$$(\hat{C}_{lp}, \hat{\rho}_{lp}) \models \text{loop}$$

we have (among others) to show

$$(\hat{C}_{lp}, \hat{\rho}_{lp}) \models (g^6 \text{ (fn } z \Rightarrow z^7)^8)^9$$

and to prove this we have (among others) to show

$$(\hat{C}_{lp}, \hat{\rho}_{lp}) \models (f^1 \text{ (fn } y \Rightarrow y^2)^3)^4$$

and to show this we have (among others) to show

$$(\hat{C}_{lp}, \hat{\rho}_{lp}) \models (f^1 \text{ (fn } y \Rightarrow y^2)^3)^4$$

because $\hat{C}_{lp}(3) \subseteq \hat{\rho}_{lp}(x)$, $\hat{C}_{lp}(4) \subseteq \hat{C}_{lp}(4)$ and $f \in \hat{\rho}_{lp}(f)$.

The Lesson

The **co-inductive definition** solves the circularity:

It allows us to assume that $(\hat{C}_{lp}, \hat{\rho}_{lp}) \models (f^1 \text{ (fn } y \Rightarrow y^2)^3)^4$ holds at the “inner level” and proving that it also holds at the “outer level”

An **inductive definition** does not give us this possibility!

Theoretical Properties:

- structural operational semantics
- semantic correctness
- the existence of least solutions

Choice of Semantics

- operational or denotational semantics?
 - an operational semantics more easily models intensional properties
- small-step or big-step operational semantics?
 - a small-step semantics allows us to reason about looping programs
- operational semantics based on environments or substitutions?
 - an environment based semantics preserves the identity of functions

Configurations and Transitions

Semantic categories:

$$v \in \mathbf{Val} \quad \text{values}$$

$$\rho \in \mathbf{Env} \quad \text{environments}$$

defined by:

$$v ::= c \mid \text{close } t \text{ in } \rho \quad \text{closures}$$

$$\rho ::= [] \mid \rho[x \mapsto v]$$

Transitions have the form

$$\rho \vdash e_1 \rightarrow e_2$$

meaning that *one step* of computation of the expression e_1 in the environment ρ will transform it into e_2 .

Transitions

$\rho \vdash x^\ell \xrightarrow{\text{green}} v^\ell$ if $x \in \text{dom}(\rho)$ and $v = \rho(x)$

$\rho \vdash (\text{fn } x \Rightarrow e_0)^\ell \xrightarrow{\text{green}} (\text{close } (\text{fn } x \Rightarrow e_0) \text{ in } \rho_0)^\ell$

where $\rho_0 = \rho \mid FV(\text{fn } x \Rightarrow e_0)$

$\rho \vdash (\text{fun } f \ x \Rightarrow e_0)^\ell \xrightarrow{\text{green}} (\text{close } (\text{fun } f \ x \Rightarrow e_0) \text{ in } \rho_0)^\ell$

where $\rho_0 = \rho \mid FV(\text{fun } f \ x \Rightarrow e_0)$

static scope!

Intermediate Expressions and Terms

$ie \in \mathbf{IExp}$ *intermediate expressions*

$it \in \mathbf{ITerm}$ *intermediate terms*

extending the syntax:

$ie ::= it^\ell$

$it ::= c \mid x \mid \text{fn } x \Rightarrow e_0 \mid \text{fun } f \ x \Rightarrow e_0 \mid ie_1 \ ie_2$
| if ie_0 then e_1 else e_2 | let $x = ie_1$ in e_2 | $ie_1 \ op \ ie_2$
| close t in ρ | bind ρ in ie

The correct form of transitions

$\rho \vdash ie_1 \rightarrow ie_2$

Transitions

$$\frac{\rho \vdash ie_1 \rightarrow ie'_1}{\rho \vdash (ie_1 \ ie_2)^\ell \rightarrow (ie'_1 \ ie_2)^\ell}$$

$$\frac{\rho \vdash ie_2 \rightarrow ie'_2}{\rho \vdash (v_1^{\ell_1} \ ie_2)^\ell \rightarrow (v_1^{\ell_1} \ ie'_2)^\ell}$$

$$\rho \vdash ((\text{close } (\text{fn } x \Rightarrow e_1) \text{ in } \rho_1)^{\ell_1} \ v_2^{\ell_2})^\ell \rightarrow (\text{bind } \rho_1[x \mapsto v_2] \text{ in } e_1)^\ell$$

$$\rho \vdash ((\text{close } (\text{fun } f \ x \Rightarrow e_1) \text{ in } \rho_1)^{\ell_1} \ v_2^{\ell_2})^\ell \rightarrow (\text{bind } \rho_2[x \mapsto v_2] \text{ in } e_1)^\ell$$

where $\rho_2 = \rho_1[f \mapsto \text{close } (\text{fun } f \ x \Rightarrow e_1) \text{ in } \rho_1]$

$$\frac{\rho_1 \vdash ie_1 \rightarrow ie'_1}{\rho \vdash (\text{bind } \rho_1 \text{ in } ie_1)^\ell \rightarrow (\text{bind } \rho_1 \text{ in } ie'_1)^\ell}$$

$$\rho \vdash (\text{bind } \rho_1 \text{ in } v_1^{\ell_1})^\ell \rightarrow v_1^\ell$$

the outermost label remains the same

Example:

$[] \vdash ((\text{fn } x \Rightarrow x^1)^2 (\text{fn } y \Rightarrow y^3)^4)^5$

$\rightarrow ((\text{close } (\text{fn } x \Rightarrow x^1) \text{ in } [])^2 (\text{fn } y \Rightarrow y^3)^4)^5$

$\rightarrow ((\text{close } (\text{fn } x \Rightarrow x^1) \text{ in } [])^2 (\text{close } (\text{fn } y \Rightarrow y^3) \text{ in } [])^4)^5$

$\rightarrow (\text{bind } [x \mapsto (\text{close } (\text{fn } y \Rightarrow y^3) \text{ in } [])] \text{ in } x^1)^5$

$\rightarrow (\text{bind } [x \mapsto (\text{close } (\text{fn } y \Rightarrow y^3) \text{ in } [])] \text{ in } \\ (\text{close } (\text{fn } y \Rightarrow y^3) \text{ in } [])^1)^5$

$\rightarrow (\text{close } (\text{fn } y \Rightarrow y^3) \text{ in } [])^5$

Transitions

$$\frac{\rho \vdash ie_0 \rightarrow ie'_0}{\rho \vdash (\text{if } ie_0 \text{ then } e_1 \text{ else } e_2)^\ell \rightarrow (\text{if } ie'_0 \text{ then } e_1 \text{ else } e_2)^\ell}$$

$$\rho \vdash (\text{if true}^{\ell_0} \text{ then } t_1^{\ell_1} \text{ else } t_2^{\ell_2})^\ell \rightarrow t_1^\ell$$

$$\rho \vdash (\text{if false}^{\ell_0} \text{ then } t_1^{\ell_1} \text{ else } t_2^{\ell_2})^\ell \rightarrow t_2^\ell$$

$$\frac{\rho \vdash ie_1 \rightarrow ie'_1}{\rho \vdash (\text{let } x = ie_1 \text{ in } e_2)^\ell \rightarrow (\text{let } x = ie'_1 \text{ in } e_2)^\ell}$$

$$\rho \vdash (\text{let } x = v^{\ell_1} \text{ in } e_2)^\ell \rightarrow (\text{bind } [x \mapsto v] \text{ in } e_2)^\ell$$

$$\frac{\rho \vdash ie_1 \rightarrow ie'_1}{\rho \vdash (ie_1 \text{ op } ie_2)^\ell \rightarrow (ie'_1 \text{ op } ie_2)^\ell}$$

$$\frac{\rho \vdash ie_2 \rightarrow ie'_2}{\rho \vdash (v_1^{\ell_1} \text{ op } ie_2)^\ell \rightarrow (v_1^{\ell_1} \text{ op } ie'_2)^\ell}$$

$$\rho \vdash (v_1^{\ell_1} \text{ op } v_2^{\ell_2})^\ell \rightarrow v^\ell \quad \text{if } v = v_1 \text{ op } v_2$$

Example:

```
[ ] ⊢ (let g = (fun f x => (f1 (fn y => y2)3)4)5
      in (g6 (fn z => z7)8)9)10
→ (let g = f5 in (g6 (fn z => z7)8)9)10
→ (bind [g ↦ f] in (g6 (fn z => z7)8)9)10
→ (bind [g ↦ f] in (f6 (fn z => z7)8)9)10
→ (bind [g ↦ f] in (f6 idz8)9)10
→ (bind [g ↦ f] in (bind [f ↦ f][x ↦ idz] in (f1 (fn y => y2)3)4)9)10
→* (bind [g ↦ f] in (bind [f ↦ f][x ↦ idz] in
      (bind [f ↦ f][x ↦ idy] in (f1 (fn y => y2)3)4)4)9)10
→* ...
```

Abbreviations:

$$\begin{aligned} f &= \text{close } (\text{fun } f \ x \Rightarrow (f^1 \ (\text{fn } y \Rightarrow y^2)^3)^4) \text{ in } [] \\ \text{id}_y &= \text{close } (\text{fn } y \Rightarrow y^2) \text{ in } [] \\ \text{id}_z &= \text{close } (\text{fn } z \Rightarrow z^7) \text{ in } [] \end{aligned}$$

Semantic Correctness

A *subject reduction result*: an acceptable result of the analysis remains acceptable under evaluation

Analysis of intermediate expressions

$$\begin{array}{l} (\hat{C}, \hat{\rho}) \models (\text{bind } \rho \text{ in } it_0^{\ell_0})^\ell \\ \text{iff} \quad (\hat{C}, \hat{\rho}) \models it_0^{\ell_0} \wedge \hat{C}(\ell_0) \subseteq \hat{C}(\ell) \wedge \rho \mathcal{R} \hat{\rho} \end{array}$$

$$\begin{array}{l} (\hat{C}, \hat{\rho}) \models (\text{close } t_0 \text{ in } \rho)^\ell \\ \text{iff} \quad \{t_0\} \subseteq \hat{C}(\ell) \wedge \rho \mathcal{R} \hat{\rho} \end{array}$$

Correctness Relation

The **global** abstract environment, $\hat{\rho}$ models *all* the **local** environments of the semantics

Correctness relation

$$\mathcal{R} : (\mathbf{Env} \times \widehat{\mathbf{Env}}) \rightarrow \{true, false\}$$

We demand that $\rho \mathcal{R} \hat{\rho}$ for all local environments, ρ , occurring in the intermediate expressions

Define

$$\rho \mathcal{R} \hat{\rho} \quad \underline{\text{iff}} \quad \forall x \in \text{dom}(\rho) \subseteq \text{dom}(\hat{\rho}) \quad \forall t_x \quad \forall \rho_x : \\ (\rho(x) = \text{close } t_x \text{ in } \rho_x) \Rightarrow (t_x \in \hat{\rho}(x) \wedge \rho_x \mathcal{R} \hat{\rho})$$

(Well-defined by induction in the size of ρ .)

Example:

Suppose that:

$$\begin{aligned}\rho &= [x \mapsto \text{close } t_1 \text{ in } \rho_1][y \mapsto \text{close } t_2 \text{ in } \rho_2] \\ \rho_1 &= [] \\ \rho_2 &= [x \mapsto \text{close } t_3 \text{ in } \rho_3] \\ \rho_3 &= []\end{aligned}$$

Then $\rho \mathcal{R} \hat{\rho}$ amounts to $\{t_1, t_3\} \subseteq \hat{\rho}(x) \wedge \{t_2\} \subseteq \hat{\rho}(y)$.

Alternative definition of Correctness Relation

Split the definition of \mathcal{R} into two components:

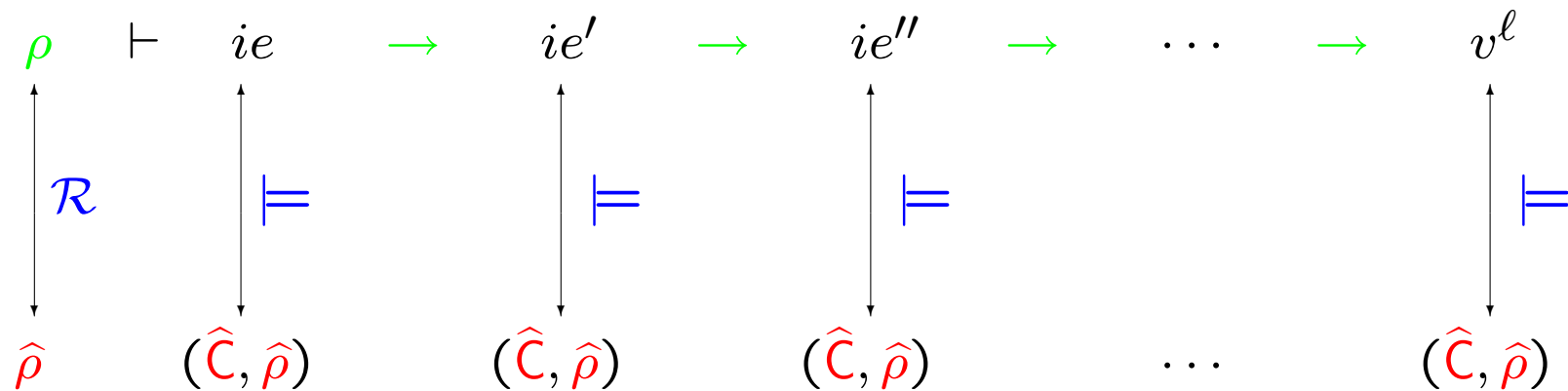
$$\mathcal{V} : (\text{Val} \times (\widehat{\text{Env}} \times \widehat{\text{Val}})) \rightarrow \{true, false\}$$

$$\mathcal{R} : (\text{Env} \times \widehat{\text{Env}}) \rightarrow \{true, false\}$$

and define

$$\begin{array}{ll} v \mathcal{V} (\hat{\rho}, \hat{v}) & \text{iff} \quad \forall t \forall \rho : (v = \text{close } t \text{ in } \rho) \Rightarrow (t \in \hat{v} \wedge \rho \mathcal{R} \hat{\rho}) \\ \rho \mathcal{R} \hat{\rho} & \text{iff} \quad \forall x \in \text{dom}(\rho) \subseteq \text{dom}(\hat{\rho}) : \rho(x) \mathcal{V} (\hat{\rho}, \hat{\rho}(x)) \end{array}$$

Correctness Result



Formal details of Correctness Result

Theorem:

If $\rho \mathcal{R} \hat{\rho}$ and $\rho \vdash ie \rightarrow ie'$ then $(\hat{C}, \hat{\rho}) \models_{ie}$ implies $(\hat{C}, \hat{\rho}) \models_{ie'}$.

Intuitively:

If there is a possible evaluation of the program such that the function at a call point evaluates to some abstraction, then this abstraction has to be in the set of possible abstractions computed by the analysis.

Observe: the theorem expresses that *all* acceptable analysis results remain acceptable under evaluation!

Thus we do *not* rely on the existence of a least or “best” solution.

Proof of Correctness Result

We assume that $\rho \mathcal{R} \hat{\rho}$ and $(\hat{\mathcal{C}}, \hat{\rho}) \models_{ie}$ and prove $(\hat{\mathcal{C}}, \hat{\rho}) \models_{ie'}$ by induction on the structure of the inference tree for $\rho \vdash ie \rightarrow ie'$.

Most cases amount to inspecting the defining clause for $(\hat{\mathcal{C}}, \hat{\rho}) \models ie$.

This method of proof applies to *all* fixed points of a recursive definition and in particular also to the (more familiar least and) greatest fixed point(s).

Crucial fact: If $(\hat{\mathcal{C}}, \hat{\rho}) \models_{it^{\ell_1}}$ and $\hat{\mathcal{C}}(\ell_1) \subseteq \hat{\mathcal{C}}(\ell_2)$ then $(\hat{\mathcal{C}}, \hat{\rho}) \models_{it^{\ell_2}}$.

Example:

Semantics:

$$[] \vdash ((\text{fn } x \Rightarrow x^1)^2 (\text{fn } y \Rightarrow y^3)^4)^5 \xrightarrow{*} (\text{close } (\text{fn } y \Rightarrow y^3) \text{ in } [])^5$$

	$(\hat{C}_e, \hat{\rho}_e)$
1	$\{\text{fn } y \Rightarrow y^3\}$
2	$\{\text{fn } x \Rightarrow x^1\}$
3	\emptyset
4	$\{\text{fn } y \Rightarrow y^3\}$
5	$\{\text{fn } y \Rightarrow y^3\}$
x	$\{\text{fn } y \Rightarrow y^3\}$
y	\emptyset

Analysis:

$$(\hat{C}_e, \hat{\rho}_e) \models ((\text{fn } x \Rightarrow x^1)^2 (\text{fn } y \Rightarrow y^3)^4)^5$$

Correctness relation:

$$[] \mathcal{R} \hat{\rho}_e$$

Correctness theorem: $(\hat{C}_e, \hat{\rho}_e) \models (\text{close } (\text{fn } y \Rightarrow y^3) \text{ in } [])^5$

Existence of Solutions

- Does each expression e admit a Control Flow Analysis?

i.e. does there exist $(\hat{\mathbf{C}}, \hat{\rho})$ such that $(\hat{\mathbf{C}}, \hat{\rho}) \models e$?

- Does each expression e have a “least” Control Flow Analysis?

i.e. does there exist $(\hat{\mathbf{C}}_0, \hat{\rho}_0)$ such that $(\hat{\mathbf{C}}_0, \hat{\rho}_0) \models e$ and such that whenever $(\hat{\mathbf{C}}, \hat{\rho}) \models e$ then $(\hat{\mathbf{C}}_0, \hat{\rho}_0)$ is “less than” $(\hat{\mathbf{C}}, \hat{\rho})$?

Here “least” is with respect to the partial ordering

$$(\hat{\mathbf{C}}_1, \hat{\rho}_1) \sqsubseteq (\hat{\mathbf{C}}_2, \hat{\rho}_2) \quad \text{iff} \quad (\forall \ell \in \mathbf{Lab} : \hat{\mathbf{C}}_1(\ell) \subseteq \hat{\mathbf{C}}_2(\ell)) \wedge (\forall x \in \mathbf{Var} : \hat{\rho}_1(x) \subseteq \hat{\rho}_2(x))$$

Existence of Solutions (cont.)

Two answers:

- there exists algorithms for the efficient computation of least solutions for all expressions
- all intermediate expressions enjoy a Moore family property

A subset Y of a complete lattice $L = (L, \sqsubseteq)$ is a *Moore family* if and only if $(\bigsqcap Y') \in Y$ for all subsets Y' of L

Proposition: The set $\{(\hat{C}, \hat{\rho}) \mid (\hat{C}, \hat{\rho}) \models ie\}$ is a Moore family for all intermediate expressions ie

Existence of Solutions (cont.)

All intermediate expressions admit a Control Flow Analysis

Let Y' be the empty set; then $\bigcap Y'$ is an element of $\{(\hat{C}, \hat{\rho}) \mid (\hat{C}, \hat{\rho}) \models ie\}$ showing that there exists at least one analysis of ie .

All intermediate expressions have a least Control Flow Analysis

Let Y' be the set $\{(\hat{C}, \hat{\rho}) \mid (\hat{C}, \hat{\rho}) \models ie\}$; then $\bigcap Y'$ is an element of $\{(\hat{C}, \hat{\rho}) \mid (\hat{C}, \hat{\rho}) \models ie\}$ so it will also be an analysis of ie . Clearly $\bigcap Y' \sqsubseteq (\hat{C}, \hat{\rho})$ for all other analyses $(\hat{C}, \hat{\rho})$ of ie so it is the least analysis result.

Example:

$$(\hat{C}_e', \hat{\rho}_e') \models ((\text{fn } x \Rightarrow x^1)^2 (\text{fn } y \Rightarrow y^3)^4)^5$$

$$(\hat{C}_e'', \hat{\rho}_e'') \models ((\text{fn } x \Rightarrow x^1)^2 (\text{fn } y \Rightarrow y^3)^4)^5$$

The Moore family result ensures that

$$(\hat{C}_e' \sqcap \hat{C}_e'', \hat{\rho}_e' \sqcap \hat{\rho}_e'') \models ((\text{fn } x \Rightarrow x^1)^2 (\text{fn } y \Rightarrow y^3)^4)^5$$

	$(\hat{C}_e, \hat{\rho}_e)$	$(\hat{C}_e', \hat{\rho}_e')$	$(\hat{C}_e'', \hat{\rho}_e'')$
1	$\{\text{fn } y \Rightarrow y^3\}$	$\{\text{fn } y \Rightarrow y^3\}$	$\{\text{fn } y \Rightarrow y^3\}$
2	$\{\text{fn } x \Rightarrow x^1\}$	$\{\text{fn } x \Rightarrow x^1\}$	$\{\text{fn } x \Rightarrow x^1\}$
3	\emptyset	$\{\text{fn } x \Rightarrow x^1\}$	$\{\text{fn } y \Rightarrow y^3\}$
4	$\{\text{fn } y \Rightarrow y^3\}$	$\{\text{fn } y \Rightarrow y^3\}$	$\{\text{fn } y \Rightarrow y^3\}$
5	$\{\text{fn } y \Rightarrow y^3\}$	$\{\text{fn } y \Rightarrow y^3\}$	$\{\text{fn } y \Rightarrow y^3\}$
x	$\{\text{fn } y \Rightarrow y^3\}$	$\{\text{fn } y \Rightarrow y^3\}$	$\{\text{fn } y \Rightarrow y^3\}$
y	\emptyset	$\{\text{fn } x \Rightarrow x^1\}$	$\{\text{fn } y \Rightarrow y^3\}$

Coinduction versus Induction

The abstract Control Flow Analysis is defined *coinductively*

\models is the *greatest* fixed point of a function Q

An alternative might be an *inductive* definition

\models' is the *least* fixed point of the function Q .

Proposition: There exists $e_\star \in \mathbf{Exp}$ such that $\{(\hat{C}, \hat{\rho}) \mid (\hat{C}, \hat{\rho}) \models' e_\star\}$ is *not* a Moore family.

Syntax Directed 0-CFA Analysis

Reformulate the abstract specification:

- (i) Syntax directed specification
- (ii) Constructing a finite set of constraints
- (iii) Compute the least solution of the set of constraints

Common Phenomenon

A specification \models_A is reformulated into a specification \models_B ensuring that

$$(\hat{C}, \hat{\rho}) \models_A e_\star \Leftarrow (\hat{C}, \hat{\rho}) \models_B e_\star$$

so that “ \models_B ” is a *safe approximation* to “ \models_A ” and hence the best (i.e. least) solution to “ $\models_B e_\star$ ” will also be a solution to “ $\models_A e_\star$ ”.

If additionally

$$(\hat{C}, \hat{\rho}) \models_A e_\star \Rightarrow (\hat{C}, \hat{\rho}) \models_B e_\star$$

then we can be assured that *no solutions are lost* and hence the best (i.e. least) solution to “ $\models_B e_\star$ ” will also be the best (i.e. least) solution to “ $\models_A e_\star$ ”.

Syntax Directed Specification (1)

$$\begin{aligned}
 (\hat{C}, \hat{\rho}) &\models_s (\text{fn } x \Rightarrow e_0)^\ell \\
 \text{iff} \quad &\{\text{fn } x \Rightarrow e_0\} \subseteq \hat{C}(\ell) \wedge \\
 &(\hat{C}, \hat{\rho}) \models_s e_0
 \end{aligned}$$

$$\begin{aligned}
 (\hat{C}, \hat{\rho}) &\models_s (\text{fun } f \ x \Rightarrow e_0)^\ell \\
 \text{iff} \quad &\{\text{fun } f \ x \Rightarrow e_0\} \subseteq \hat{C}(\ell) \wedge \\
 &(\hat{C}, \hat{\rho}) \models_s e_0 \wedge \{\text{fun } f \ x \Rightarrow e_0\} \subseteq \hat{\rho}(f)
 \end{aligned}$$

$$\begin{aligned}
 (\hat{C}, \hat{\rho}) &\models_s (t_1^{\ell_1} \ t_2^{\ell_2})^\ell \\
 \text{iff} \quad &(\hat{C}, \hat{\rho}) \models_s t_1^{\ell_1} \wedge (\hat{C}, \hat{\rho}) \models_s t_2^{\ell_2} \wedge \\
 &(\forall (\text{fn } x \Rightarrow t_0^{\ell_0}) \in \hat{C}(\ell_1) : \\
 &\quad \hat{C}(\ell_2) \subseteq \hat{\rho}(x) \wedge \hat{C}(\ell_0) \subseteq \hat{C}(\ell) \quad \boxed{}) \wedge \\
 &(\forall (\text{fun } f \ x \Rightarrow t_0^{\ell_0}) \in \hat{C}(\ell_1) : \\
 &\quad \hat{C}(\ell_2) \subseteq \hat{\rho}(x) \wedge \hat{C}(\ell_0) \subseteq \hat{C}(\ell) \quad \boxed{})
 \end{aligned}$$

Syntax Directed Specification (2)

$$(\hat{C}, \hat{\rho}) \models_s c^\ell \text{ always}$$

$$(\hat{C}, \hat{\rho}) \models_s x^\ell \quad \underline{\text{iff}} \quad \hat{\rho}(x) \subseteq \hat{C}(\ell)$$

$$\begin{aligned} (\hat{C}, \hat{\rho}) \models_s (\text{if } t_0^{\ell_0} \text{ then } t_1^{\ell_1} \text{ else } t_2^{\ell_2})^\ell \\ \underline{\text{iff}} \quad & (\hat{C}, \hat{\rho}) \models_s t_0^{\ell_0} \wedge \\ & (\hat{C}, \hat{\rho}) \models_s t_1^{\ell_1} \wedge (\hat{C}, \hat{\rho}) \models_s t_2^{\ell_2} \wedge \\ & \hat{C}(\ell_1) \subseteq \hat{C}(\ell) \wedge \hat{C}(\ell_2) \subseteq \hat{C}(\ell) \end{aligned}$$

$$\begin{aligned} (\hat{C}, \hat{\rho}) \models_s (\text{let } x = t_1^{\ell_1} \text{ in } t_2^{\ell_2})^\ell \\ \underline{\text{iff}} \quad & (\hat{C}, \hat{\rho}) \models_s t_1^{\ell_1} \wedge (\hat{C}, \hat{\rho}) \models_s t_2^{\ell_2} \wedge \\ & \hat{C}(\ell_1) \subseteq \hat{\rho}(x) \wedge \hat{C}(\ell_2) \subseteq \hat{C}(\ell) \end{aligned}$$

$$(\hat{C}, \hat{\rho}) \models_s (t_1^{\ell_1} \text{ op } t_2^{\ell_2})^\ell \quad \underline{\text{iff}} \quad (\hat{C}, \hat{\rho}) \models_s t_1^{\ell_1} \wedge (\hat{C}, \hat{\rho}) \models_s t_2^{\ell_2}$$

Example: loop

```
(let g = (fun f x => (f1 (fn y => y2)3)4)5
  in (g6 (fn z => z7)8)9)10
```

Abbreviations:

$$\begin{aligned} f &= \text{fun } f \ x \Rightarrow (f^1 \ (\text{fn } y \Rightarrow y^2)^3)^4 \\ \text{id}_y &= \text{fn } y \Rightarrow y^2 \\ \text{id}_z &= \text{fn } z \Rightarrow z^7 \end{aligned}$$

One guess of a 0-CFA analysis result:

$\hat{C}_{lp}(1) = \{f\}$	$\hat{C}_{lp}(6) = \{f\}$	$\hat{\rho}_{lp}(f) = \{f\}$
$\hat{C}_{lp}(2) = \emptyset$	$\hat{C}_{lp}(7) = \emptyset$	$\hat{\rho}_{lp}(g) = \{f\}$
$\hat{C}_{lp}(3) = \{\text{id}_y\}$	$\hat{C}_{lp}(8) = \{\text{id}_z\}$	$\hat{\rho}_{lp}(x) = \{\text{id}_y, \text{id}_z\}$
$\hat{C}_{lp}(4) = \emptyset$	$\hat{C}_{lp}(9) = \emptyset$	$\hat{\rho}_{lp}(y) = \emptyset$
$\hat{C}_{lp}(5) = \{f\}$	$\hat{C}_{lp}(10) = \emptyset$	$\hat{\rho}_{lp}(z) = \emptyset$

Example: Checking the solution

To show

$$(\hat{C}_{lp}, \hat{\rho}_{lp}) \models_s \text{loop}$$

we have (among others) to show

$$(\hat{C}_{lp}, \hat{\rho}_{lp}) \models_s (g^6 \text{ (fn } z \Rightarrow z^7)^8)^9$$

and

$$(\hat{C}_{lp}, \hat{\rho}_{lp}) \models_s (f^1 \text{ (fn } y \Rightarrow y^2)^3)^4$$

and this is straightforward.

The Lesson

No need for **co-induction** because the definition is syntax-directed

Preservation of Solutions

Define $(\hat{\mathbf{C}}_\star^\top, \hat{\rho}_\star^\top)$ by:

$$\hat{\mathbf{C}}_\star^\top(\ell) = \begin{cases} \emptyset & \text{if } \ell \notin \mathbf{Lab}_\star \\ \mathbf{Term}_\star & \text{if } \ell \in \mathbf{Lab}_\star \end{cases}$$

$$\hat{\rho}_\star^\top(x) = \begin{cases} \emptyset & \text{if } x \notin \mathbf{Var}_\star \\ \mathbf{Term}_\star & \text{if } x \in \mathbf{Var}_\star \end{cases}$$

Then all the solutions to “ $\models_s e_\star$ ” that are “less than” $(\hat{\mathbf{C}}_\star^\top, \hat{\rho}_\star^\top)$ are solutions to “ $\models e_\star$ ” as well:

Proposition: If $(\hat{\mathbf{C}}, \hat{\rho}) \models_s e_\star$ and $(\hat{\mathbf{C}}, \hat{\rho}) \sqsubseteq (\hat{\mathbf{C}}_\star^\top, \hat{\rho}_\star^\top)$ then $(\hat{\mathbf{C}}, \hat{\rho}) \models e_\star$.

(That $(\hat{\mathbf{C}}, \hat{\rho}) \sqsubseteq (\hat{\mathbf{C}}_\star^\top, \hat{\rho}_\star^\top)$ means that $(\hat{\mathbf{C}}, \hat{\rho})$ lives in a “closed universe”.)

Proposition:

$\{(\hat{C}, \hat{\rho}) \sqsubseteq (\hat{C}_\star^\top, \hat{\rho}_\star^\top) \mid (\hat{C}, \hat{\rho}) \models_s e_\star\}$ is a Moore family.

Corollaries:

- each expression e_\star has a Control Flow Analysis that is “less than” $(\hat{C}_\star^\top, \hat{\rho}_\star^\top)$, and
- each expression e_\star has a “least” Control Flow Analysis that is “less than” $(\hat{C}_\star^\top, \hat{\rho}_\star^\top)$.

Constraint Based 0-CFA Analysis

$\mathcal{C}_\star[[e_\star]]$ is a set of constraints of the form

$$lhs \subseteq rhs$$

$$\{t\} \subseteq rhs' \Rightarrow lhs \subseteq rhs$$

where

$$rhs ::= C(\ell) \mid r(x)$$

$$lhs ::= C(\ell) \mid r(x) \mid \{t\}$$

and all occurrences of t are of the form `fn $x \Rightarrow e_0$` or `fun $f \ x \Rightarrow e_0$`

Constraint Based Control Flow Analysis (1)

$$\mathcal{C}_\star[\llbracket (\text{fn } x \Rightarrow e_0)^\ell \rrbracket] = \{ \{ \text{fn } x \Rightarrow e_0 \} \subseteq \mathcal{C}(\ell) \} \cup \mathcal{C}_\star[\llbracket e_0 \rrbracket]$$

$$\begin{aligned} \mathcal{C}_\star[\llbracket (\text{fun } f \ x \Rightarrow e_0)^\ell \rrbracket] &= \{ \{ \text{fun } f \ x \Rightarrow e_0 \} \subseteq \mathcal{C}(\ell) \} \cup \mathcal{C}_\star[\llbracket e_0 \rrbracket] \\ &\cup \{ \{ \text{fun } f \ x \Rightarrow e_0 \} \subseteq r(f) \} \end{aligned}$$

$$\begin{aligned} \mathcal{C}_\star[\llbracket (t_1^{\ell_1} \ t_2^{\ell_2})^\ell \rrbracket] &= \mathcal{C}_\star[\llbracket t_1^{\ell_1} \rrbracket] \cup \mathcal{C}_\star[\llbracket t_2^{\ell_2} \rrbracket] \\ &\cup \{ \{t\} \subseteq \mathcal{C}(\ell_1) \Rightarrow \mathcal{C}(\ell_2) \subseteq r(x) \mid t = (\text{fn } x \Rightarrow t_0^{\ell_0}) \in \mathbf{Term}_\star \} \\ &\cup \{ \{t\} \subseteq \mathcal{C}(\ell_1) \Rightarrow \mathcal{C}(\ell_0) \subseteq \mathcal{C}(\ell) \mid t = (\text{fn } x \Rightarrow t_0^{\ell_0}) \in \mathbf{Term}_\star \} \\ &\cup \{ \{t\} \subseteq \mathcal{C}(\ell_1) \Rightarrow \mathcal{C}(\ell_2) \subseteq r(x) \mid t = (\text{fun } f \ x \Rightarrow t_0^{\ell_0}) \in \mathbf{Term}_\star \} \\ &\cup \{ \{t\} \subseteq \mathcal{C}(\ell_1) \Rightarrow \mathcal{C}(\ell_0) \subseteq \mathcal{C}(\ell) \mid t = (\text{fun } f \ x \Rightarrow t_0^{\ell_0}) \in \mathbf{Term}_\star \} \end{aligned}$$

(Eager rather than lazy unfolding – easy but costly.)

Constraint Based Control Flow Analysis (2)

$$C_{\star}[[c^{\ell}]] = \emptyset$$

$$C_{\star}[[x^{\ell}]] = \{ r(x) \subseteq C(\ell) \}$$

$$\begin{aligned} C_{\star}[(\text{if } t_0^{\ell_0} \text{ then } t_1^{\ell_1} \text{ else } t_2^{\ell_2})^{\ell}] &= C_{\star}[[t_0^{\ell_0}]] \cup C_{\star}[[t_1^{\ell_1}]] \cup C_{\star}[[t_2^{\ell_2}]] \\ &\quad \cup \{ C(\ell_1) \subseteq C(\ell) \} \\ &\quad \cup \{ C(\ell_2) \subseteq C(\ell) \} \end{aligned}$$

$$\begin{aligned} C_{\star}[(\text{let } x = t_1^{\ell_1} \text{ in } t_2^{\ell_2})^{\ell}] &= C_{\star}[[t_1^{\ell_1}]] \cup C_{\star}[[t_2^{\ell_2}]] \\ &\quad \cup \{ C(\ell_1) \subseteq r(x) \} \cup \{ C(\ell_2) \subseteq C(\ell) \} \end{aligned}$$

$$C_{\star}[(t_1^{\ell_1} \text{ op } t_2^{\ell_2})^{\ell}] = C_{\star}[[t_1^{\ell_1}]] \cup C_{\star}[[t_2^{\ell_2}]]$$

Example:

$$\begin{aligned} \mathcal{C}_\star \llbracket ((\text{fn } x \Rightarrow x^1)^2 \ (\text{fn } y \Rightarrow y^3)^4)^5 \rrbracket = \\ \{ \{ \text{fn } x \Rightarrow x^1 \} \subseteq \mathcal{C}(2), \\ r(x) \subseteq \mathcal{C}(1), \\ \{ \text{fn } y \Rightarrow y^3 \} \subseteq \mathcal{C}(4), \\ r(y) \subseteq \mathcal{C}(3), \\ \{ \text{fn } x \Rightarrow x^1 \} \subseteq \mathcal{C}(2) \Rightarrow \mathcal{C}(4) \subseteq r(x), \\ \{ \text{fn } x \Rightarrow x^1 \} \subseteq \mathcal{C}(2) \Rightarrow \mathcal{C}(1) \subseteq \mathcal{C}(5), \\ \{ \text{fn } y \Rightarrow y^3 \} \subseteq \mathcal{C}(2) \Rightarrow \mathcal{C}(4) \subseteq r(y), \\ \{ \text{fn } y \Rightarrow y^3 \} \subseteq \mathcal{C}(2) \Rightarrow \mathcal{C}(3) \subseteq \mathcal{C}(5) \} \end{aligned}$$

Preservation of Solutions

Translating syntactic entities to sets of terms:

$$\begin{aligned}(\hat{C}, \hat{\rho}) \llbracket C(\ell) \rrbracket &= \hat{C}(\ell) \\(\hat{C}, \hat{\rho}) \llbracket r(x) \rrbracket &= \hat{\rho}(x) \\(\hat{C}, \hat{\rho}) \llbracket \{t\} \rrbracket &= \{t\}\end{aligned}$$

Satisfaction relation for constraints: $(\hat{C}, \hat{\rho}) \models_c (lhs \subseteq rhs)$

$$\begin{aligned}(\hat{C}, \hat{\rho}) \models_c (lhs \subseteq rhs) \\ \text{iff } (\hat{C}, \hat{\rho}) \llbracket lhs \rrbracket \subseteq (\hat{C}, \hat{\rho}) \llbracket rhs \rrbracket\end{aligned}$$

$$\begin{aligned}(\hat{C}, \hat{\rho}) \models_c (\{t\} \subseteq rhs' \Rightarrow lhs \subseteq rhs) \\ \text{iff } (\{t\} \subseteq (\hat{C}, \hat{\rho}) \llbracket rhs' \rrbracket \wedge (\hat{C}, \hat{\rho}) \llbracket lhs \rrbracket \subseteq (\hat{C}, \hat{\rho}) \llbracket rhs \rrbracket) \\ \vee (\{t\} \not\subseteq (\hat{C}, \hat{\rho}) \llbracket rhs' \rrbracket)\end{aligned}$$

Proposition: $(\hat{C}, \hat{\rho}) \models_s e_\star$ if and only if $(\hat{C}, \hat{\rho}) \models_c \mathcal{C}_\star \llbracket e_\star \rrbracket$.

Solving the Constraints (1)

Input: a set of constraints $\mathcal{C}_\star[[e_\star]]$

Output: the least solution $(\hat{C}, \hat{\rho})$ to the constraints

Data structures: a graph with one node for each $C(\ell)$ and $r(x)$ (where $\ell \in \mathbf{Lab}_\star$ and $x \in \mathbf{Var}_\star$) and zero, one or two edges for each constraint in $\mathcal{C}_\star[[e_\star]]$

- **W**: the worklist of the nodes whose outgoing edges should be traversed
- **D**: an array that for each node gives an element of $\widehat{\mathbf{Val}}_\star$
- **E**: an array that for each node gives a list of constraints influenced (and outgoing edges)

Auxiliary procedure:

procedure $\text{add}(q, d)$ **is** if $\neg (d \subseteq D[q])$ then $D[q] := D[q] \cup d;$
 $W := \text{cons}(q, W);$

Solving the Constraints (2)

Step 1 Initialisation

$W := \text{nil};$
for q in Nodes do $D[q] := \emptyset; E[q] := \text{nil};$

Step 2 Building the graph

for cc in $\mathcal{C}_\star[[e_\star]]$ do
case cc of $\{t\} \subseteq p$: $\text{add}(p, \{t\});$
 $p_1 \subseteq p_2$: $E[p_1] := \text{cons}(cc, E[p_1]);$
 $\{t\} \subseteq p \Rightarrow p_1 \subseteq p_2$: $E[p_1] := \text{cons}(cc, E[p_1]);$
 $E[p] := \text{cons}(cc, E[p]);$

Step 3 Iteration

while $W \neq \text{nil}$ do
 $q := \text{head}(W); W := \text{tail}(W);$
for cc in $E[q]$ do
case cc of $p_1 \subseteq p_2$: $\text{add}(p_2, D[p_1]);$
 $\{t\} \subseteq p \Rightarrow p_1 \subseteq p_2$: if $t \in D[p]$ then $\text{add}(p_2, D[p_1]);$

Step 4 Recording the solution

for ℓ in Lab_\star do $\hat{C}(\ell) := D[C(\ell)];$ for x in Var_\star do $\hat{\rho}(x) := D[r(x)];$

Example:

Initialisation of data structures

p	$D[p]$	$E[p]$
$C(1)$	\emptyset	$[id_x \subseteq C(2) \Rightarrow C(1) \subseteq C(5)]$
$C(2)$	id_x	$[id_y \subseteq C(2) \Rightarrow C(3) \subseteq C(5), \quad id_y \subseteq C(2) \Rightarrow C(4) \subseteq r(y),$ $id_x \subseteq C(2) \Rightarrow C(1) \subseteq C(5), \quad id_x \subseteq C(2) \Rightarrow C(4) \subseteq r(x)]$
$C(3)$	\emptyset	$[id_y \subseteq C(2) \Rightarrow C(3) \subseteq C(5)]$
$C(4)$	id_y	$[id_y \subseteq C(2) \Rightarrow C(4) \subseteq r(y), \quad id_x \subseteq C(2) \Rightarrow C(4) \subseteq r(x)]$
$C(5)$	\emptyset	$[]$
$r(x)$	\emptyset	$[r(x) \subseteq C(1)]$
$r(y)$	\emptyset	$[r(y) \subseteq C(3)]$

Example:

Iteration steps

W	$[C(4), C(2)]$	$[r(x), C(2)]$	$[C(1), C(2)]$	$[C(5), C(2)]$	$[C(2)]$	$[]$
p	$D[p]$	$D[p]$	$D[p]$	$D[p]$	$D[p]$	$D[p]$
C(1)	\emptyset	\emptyset	id_y	id_y	id_y	id_y
C(2)	id_x	id_x	id_x	id_x	id_x	id_x
C(3)	\emptyset	\emptyset	\emptyset	\emptyset	\emptyset	\emptyset
C(4)	id_y	id_y	id_y	id_y	id_y	id_y
C(5)	\emptyset	\emptyset	\emptyset	id_y	id_y	id_y
$r(x)$	\emptyset	id_y	id_y	id_y	id_y	id_y
$r(y)$	\emptyset	\emptyset	\emptyset	\emptyset	\emptyset	\emptyset

Correctness:

Given input $\mathcal{C}_\star[[e_\star]]$ the worklist algorithm terminates and the result $(\hat{\mathcal{C}}, \hat{\rho})$ produced by the algorithm satisfies

$$(\hat{\mathcal{C}}, \hat{\rho}) = \bigsqcap \{(\hat{\mathcal{C}}', \hat{\rho}') \mid (\hat{\mathcal{C}}', \hat{\rho}') \models_c \mathcal{C}_\star[[e_\star]]\}$$

and hence it is the least solution to $\mathcal{C}_\star[[e_\star]]$.

Complexity:

The algorithm takes at most $O(n^3)$ steps if the original expression e_\star has size n .

Adding Data Flow Analysis

Idea: extend the set $\widehat{\text{Val}}$ to contain other abstract values than just abstractions

- powerset (possibly finite)
- complete lattice (possibly satisfying Ascending Chain Condition)

Abstract Values as Powersets

Let **Data** be a set of *abstract data values* (i.e. abstract properties of booleans and integers)

$$\hat{v} \in \widehat{\text{Val}}_d = \mathcal{P}(\text{Term} \cup \mathbf{Data}) \quad \text{abstract values}$$

For each constant $c \in \mathbf{Const}$ we need an element $d_c \in \mathbf{Data}$

For each operator $op \in \mathbf{Op}$ we need a total function

$$\widehat{op} : \widehat{\text{Val}}_d \times \widehat{\text{Val}}_d \rightarrow \widehat{\text{Val}}_d$$

typically

$$\hat{v}_1 \widehat{op} \hat{v}_2 = \bigcup \{d_{op}(d_1, d_2) \mid d_1 \in \hat{v}_1 \cap \mathbf{Data}, d_2 \in \hat{v}_2 \cap \mathbf{Data}\}$$

for some $d_{op} : \mathbf{Data} \times \mathbf{Data} \rightarrow \mathcal{P}(\mathbf{Data})$

Example: *Detection of Signs Analysis*

$$\mathbf{Data}_{\text{sign}} = \{\text{tt}, \text{ff}, -, 0, +\}$$

$$d_{\text{true}} = \text{tt}$$

$$d_7 = +$$

$\widehat{+}$ is defined from

d_+	tt	ff	-	0	+
tt	\emptyset	\emptyset	\emptyset	\emptyset	\emptyset
ff	\emptyset	\emptyset	\emptyset	\emptyset	\emptyset
-	\emptyset	\emptyset	$\{-\}$	$\{-\}$	$\{-, 0, +\}$
0	\emptyset	\emptyset	$\{-\}$	$\{0\}$	$\{+\}$
+	\emptyset	\emptyset	$\{-, 0, +\}$	$\{+\}$	$\{+\}$

Abstract Values as Powersets (1)

$$(\hat{\mathbf{C}}, \hat{\rho}) \models_d (\text{fn } x \Rightarrow e_0)^\ell \quad \underline{\text{iff}} \quad \{\text{fn } x \Rightarrow e_0\} \subseteq \hat{\mathbf{C}}(\ell)$$

$$(\hat{\mathbf{C}}, \hat{\rho}) \models_d (\text{fun } f \ x \Rightarrow e_0)^\ell \quad \underline{\text{iff}} \quad \{\text{fun } f \ x \Rightarrow e_0\} \subseteq \hat{\mathbf{C}}(\ell)$$

$$\begin{aligned} (\hat{\mathbf{C}}, \hat{\rho}) \models_d (t_1^{\ell_1} \ t_2^{\ell_2})^\ell \\ \underline{\text{iff}} \quad & (\hat{\mathbf{C}}, \hat{\rho}) \models_d t_1^{\ell_1} \wedge (\hat{\mathbf{C}}, \hat{\rho}) \models_d t_2^{\ell_2} \wedge \\ & (\forall (\text{fn } x \Rightarrow t_0^{\ell_0}) \in \hat{\mathbf{C}}(\ell_1) : \\ & \quad (\hat{\mathbf{C}}, \hat{\rho}) \models_d t_0^{\ell_0} \wedge \\ & \quad \hat{\mathbf{C}}(\ell_2) \subseteq \hat{\rho}(x) \wedge \hat{\mathbf{C}}(\ell_0) \subseteq \hat{\mathbf{C}}(\ell)) \wedge \\ & (\forall (\text{fun } f \ x \Rightarrow t_0^{\ell_0}) \in \hat{\mathbf{C}}(\ell_1) : \\ & \quad (\hat{\mathbf{C}}, \hat{\rho}) \models_d t_0^{\ell_0} \wedge \\ & \quad \hat{\mathbf{C}}(\ell_2) \subseteq \hat{\rho}(x) \wedge \hat{\mathbf{C}}(\ell_0) \subseteq \hat{\mathbf{C}}(\ell) \wedge \\ & \quad \{\text{fun } f \ x \Rightarrow t_0^{\ell_0}\} \subseteq \hat{\rho}(f)) \end{aligned}$$

Abstract Values as Powersets (2)

$$(\hat{C}, \hat{\rho}) \models_d c^\ell \quad \text{iff} \quad \{d_c\} \subseteq \hat{C}(\ell)$$

$$(\hat{C}, \hat{\rho}) \models_d x^\ell \quad \text{iff} \quad \hat{\rho}(x) \subseteq \hat{C}(\ell)$$

$$(\hat{C}, \hat{\rho}) \models_d (\text{if } t_0^{\ell_0} \text{ then } t_1^{\ell_1} \text{ else } t_2^{\ell_2})^\ell$$

$$\text{iff} \quad (\hat{C}, \hat{\rho}) \models_d t_0^{\ell_0} \wedge$$

$$(d_{\text{true}} \in \hat{C}(\ell_0) \Rightarrow ((\hat{C}, \hat{\rho}) \models_d t_1^{\ell_1} \wedge \hat{C}(\ell_1) \subseteq \hat{C}(\ell))) \wedge$$

$$(d_{\text{false}} \in \hat{C}(\ell_0) \Rightarrow ((\hat{C}, \hat{\rho}) \models_d t_2^{\ell_2} \wedge \hat{C}(\ell_2) \subseteq \hat{C}(\ell)))$$

$$(\hat{C}, \hat{\rho}) \models_d (\text{let } x = t_1^{\ell_1} \text{ in } t_2^{\ell_2})^\ell$$

$$\text{iff} \quad (\hat{C}, \hat{\rho}) \models_d t_1^{\ell_1} \wedge (\hat{C}, \hat{\rho}) \models_d t_2^{\ell_2} \wedge \hat{C}(\ell_1) \subseteq \hat{\rho}(x) \wedge \hat{C}(\ell_2) \subseteq \hat{C}(\ell)$$

$$(\hat{C}, \hat{\rho}) \models_d (t_1^{\ell_1} \text{ op } t_2^{\ell_2})^\ell$$

$$\text{iff} \quad (\hat{C}, \hat{\rho}) \models_d t_1^{\ell_1} \wedge (\hat{C}, \hat{\rho}) \models_d t_2^{\ell_2} \wedge \hat{C}(\ell_1) \hat{\text{op}} \hat{C}(\ell_2) \subseteq \hat{C}(\ell)$$

Example:

```
(let f = (fn x => (if (x1 > 02)3 then (fn y => y4)5  
              else (fn z => 256)7)8)9  
  in ((f10 311)12 013)14)15
```

A pure 0-CFA analysis will not be able to discover that the `else`-branch of the conditional will never be executed.

When we combine the analysis with a Detection of Signs Analysis then the analysis can determine that only `fn y => y4` is a possible abstraction at label 12.

Example:

	$(\hat{\mathcal{C}}, \hat{\rho})$	$(\hat{\mathcal{C}}, \hat{\rho})$
1	\emptyset	$\{+\}$
2	\emptyset	$\{0\}$
3	\emptyset	$\{tt\}$
4	\emptyset	$\{0\}$
5	$\{\text{fn } y \Rightarrow y^4\}$	$\{\text{fn } y \Rightarrow y^4\}$
6	\emptyset	\emptyset
7	$\{\text{fn } z \Rightarrow 25^6\}$	\emptyset
8	$\{\text{fn } y \Rightarrow y^4, \text{fn } z \Rightarrow 25^6\}$	$\{\text{fn } y \Rightarrow y^4\}$
9	$\{\text{fn } x \Rightarrow (\dots)^8\}$	$\{\text{fn } x \Rightarrow (\dots)^8\}$
10	$\{\text{fn } x \Rightarrow (\dots)^8\}$	$\{\text{fn } x \Rightarrow (\dots)^8\}$
11	\emptyset	$\{+\}$
12	$\{\text{fn } y \Rightarrow y^4, \text{fn } z \Rightarrow 25^6\}$	$\{\text{fn } y \Rightarrow y^4\}$
13	\emptyset	$\{0\}$
14	\emptyset	$\{0\}$
15	\emptyset	$\{0\}$
f	$\{\text{fn } x \Rightarrow (\dots)^8\}$	$\{\text{fn } x \Rightarrow (\dots)^8\}$
x	\emptyset	$\{+\}$
y	\emptyset	$\{0\}$
z	\emptyset	\emptyset

Abstract Values as Complete Lattices

A *monotone structure* consists of:

- a complete lattice L , and
- a set \mathcal{F} of monotone functions of $L \times L \rightarrow L$.

An *instance* of a monotone structure consists of the structure (L, \mathcal{F}) and

- a mapping ι . from the constants $c \in \mathbf{Const}$ to values in L , and
- a mapping f . from the binary operators $op \in \mathbf{Op}$ to functions of \mathcal{F} .

Example:

A monotone structure corresponding to the previous development will have L to be $\mathcal{P}(\mathbf{Data})$ and \mathcal{F} to be the monotone functions of $\mathcal{P}(\mathbf{Data}) \times \mathcal{P}(\mathbf{Data}) \rightarrow \mathcal{P}(\mathbf{Data})$.

(L satisfies the Ascending Chain Property iff \mathbf{Data} is finite.)

An instance of the monotone structure is then obtained by taking

$$\iota_c = \{d_c\}$$

for all constants c (and with $d_c \in \mathbf{Data}$ as above) and

$$f_{op}(l_1, l_2) = \bigcup \{d_{op}(d_1, d_2) \mid d_1 \in l_1, d_2 \in l_2\}$$

for all binary operators op (and where $d_{op} : \mathbf{Data} \times \mathbf{Data} \rightarrow \mathcal{P}(\mathbf{Data})$ is as above).

Example: A monotone structure for *Constant Propagation Analysis* will have L to be $\mathbf{Z}_{\perp}^{\top} \times \mathcal{P}(\{\text{tt}, \text{ff}\})$ and \mathcal{F} to be the monotone functions of $L \times L \rightarrow L$.

An instance of the monotone structure is obtained by taking e.g. $\iota_7 = (7, \emptyset)$ and $\iota_{\text{true}} = (\perp, \{\text{tt}\})$. For a binary operator as $+$ we can take:

$$f_{+}(l_1, l_2) = \begin{cases} (z_1 + z_2, \emptyset) & \text{if } l_1 = (z_1, \dots), l_2 = (z_2, \dots), \\ & \text{and } z_1, z_2 \in \mathbf{Z} \\ (\perp, \emptyset) & \text{if } l_1 = (z_1, \dots), l_2 = (z_2, \dots), \\ & \text{and } z_1 = \perp \text{ or } z_2 = \perp \\ (\top, \emptyset) & \text{otherwise} \end{cases}$$

Abstract Domains

For the Control Flow Analysis:

$$\begin{aligned}\hat{v} &\in \widehat{\mathbf{Val}} &= \mathcal{P}(\mathbf{Term}) && \text{abstract values} \\ \hat{\rho} &\in \widehat{\mathbf{Env}} &= \mathbf{Var} \rightarrow \widehat{\mathbf{Val}} && \text{abstract environments} \\ \hat{C} &\in \widehat{\mathbf{Cache}} &= \mathbf{Lab} \rightarrow \widehat{\mathbf{Val}} && \text{abstract caches}\end{aligned}$$

For the Data Flow Analysis:

$$\begin{aligned}\hat{d} &\in \widehat{\mathbf{Data}} &= L && \text{abstract data values} \\ \hat{\delta} &\in \widehat{\mathbf{DEnv}} &= \mathbf{Var} \rightarrow \widehat{\mathbf{Data}} && \text{abstract data environments} \\ \hat{D} &\in \widehat{\mathbf{DCache}} &= \mathbf{Lab} \rightarrow \widehat{\mathbf{Data}} && \text{abstract data caches}\end{aligned}$$

Abstract Values as Complete Lattices (1)

$$(\hat{\mathbf{C}}, \hat{\mathbf{D}}, \hat{\rho}, \hat{\delta}) \models_D (\text{fn } x \Rightarrow e_0)^\ell \quad \text{iff} \quad \{\text{fn } x \Rightarrow e_0\} \subseteq \hat{\mathbf{C}}(\ell)$$

$$(\hat{\mathbf{C}}, \hat{\mathbf{D}}, \hat{\rho}, \hat{\delta}) \models_D (\text{fun } f \ x \Rightarrow e_0)^\ell \quad \text{iff} \quad \{\text{fun } f \ x \Rightarrow e_0\} \subseteq \hat{\mathbf{C}}(\ell)$$

$$\begin{aligned} (\hat{\mathbf{C}}, \hat{\mathbf{D}}, \hat{\rho}, \hat{\delta}) \models_D (t_1^{\ell_1} \ t_2^{\ell_2})^\ell & \\ \text{iff} \quad & (\hat{\mathbf{C}}, \hat{\mathbf{D}}, \hat{\rho}, \hat{\delta}) \models_D t_1^{\ell_1} \wedge (\hat{\mathbf{C}}, \hat{\mathbf{D}}, \hat{\rho}, \hat{\delta}) \models_D t_2^{\ell_2} \wedge \\ & (\forall (\text{fn } x \Rightarrow t_0^{\ell_0}) \in \hat{\mathbf{C}}(\ell_1) : (\hat{\mathbf{C}}, \hat{\mathbf{D}}, \hat{\rho}, \hat{\delta}) \models_D t_0^{\ell_0} \wedge \\ & \quad \hat{\mathbf{C}}(\ell_2) \subseteq \hat{\rho}(x) \wedge \hat{\mathbf{D}}(\ell_2) \sqsubseteq \hat{\delta}(x) \wedge \\ & \quad \hat{\mathbf{C}}(\ell_0) \subseteq \hat{\mathbf{C}}(\ell) \wedge \hat{\mathbf{D}}(\ell_0) \sqsubseteq \hat{\mathbf{D}}(\ell)) \wedge \\ & (\forall (\text{fun } f \ x \Rightarrow t_0^{\ell_0}) \in \hat{\mathbf{C}}(\ell_1) : (\hat{\mathbf{C}}, \hat{\mathbf{D}}, \hat{\rho}, \hat{\delta}) \models_D t_0^{\ell_0} \wedge \\ & \quad \hat{\mathbf{C}}(\ell_2) \subseteq \hat{\rho}(x) \wedge \hat{\mathbf{D}}(\ell_2) \sqsubseteq \hat{\delta}(x) \wedge \\ & \quad \hat{\mathbf{C}}(\ell_0) \subseteq \hat{\mathbf{C}}(\ell) \wedge \hat{\mathbf{D}}(\ell_0) \sqsubseteq \hat{\mathbf{D}}(\ell) \wedge \\ & \quad \{\text{fun } f \ x \Rightarrow t_0^{\ell_0}\} \subseteq \hat{\rho}(f)) \end{aligned}$$

Abstract Values as Complete Lattices (2)

$$(\hat{C}, \hat{D}, \hat{\rho}, \hat{\delta}) \models_D c^\ell \quad \text{iff} \quad \nu_C \sqsubseteq \hat{D}(\ell)$$

$$(\hat{C}, \hat{D}, \hat{\rho}, \hat{\delta}) \models_D x^\ell \quad \text{iff} \quad \hat{\rho}(x) \subseteq \hat{C}(\ell) \wedge \hat{\delta}(x) \sqsubseteq \hat{D}(\ell)$$

$$\begin{aligned} (\hat{C}, \hat{D}, \hat{\rho}, \hat{\delta}) \models_D (\text{if } t_0^{\ell_0} \text{ then } t_1^{\ell_1} \text{ else } t_2^{\ell_2})^\ell \\ \text{iff} \quad & (\hat{C}, \hat{D}, \hat{\rho}, \hat{\delta}) \models_D t_0^{\ell_0} \wedge \\ & (\nu_{\text{true}} \sqsubseteq \hat{D}(\ell_0) \Rightarrow (\hat{C}, \hat{D}, \hat{\rho}, \hat{\delta}) \models_D t_1^{\ell_1} \wedge \\ & \quad \hat{C}(\ell_1) \subseteq \hat{C}(\ell) \wedge \hat{D}(\ell_1) \sqsubseteq \hat{D}(\ell)) \wedge \\ & (\nu_{\text{false}} \sqsubseteq \hat{D}(\ell_0) \Rightarrow (\hat{C}, \hat{D}, \hat{\rho}, \hat{\delta}) \models_D t_2^{\ell_2} \wedge \\ & \quad \hat{C}(\ell_2) \subseteq \hat{C}(\ell) \wedge \hat{D}(\ell_2) \sqsubseteq \hat{D}(\ell)) \end{aligned}$$

Abstract Values as Complete Lattices (3)

$$\begin{aligned}
 (\hat{\mathbf{C}}, \hat{\mathbf{D}}, \hat{\rho}, \hat{\delta}) \models_D (\text{let } x = t_1^{\ell_1} \text{ in } t_2^{\ell_2})^\ell \\
 \text{iff } & (\hat{\mathbf{C}}, \hat{\mathbf{D}}, \hat{\rho}, \hat{\delta}) \models_D t_1^{\ell_1} \wedge \\
 & (\hat{\mathbf{C}}, \hat{\mathbf{D}}, \hat{\rho}, \hat{\delta}) \models_D t_2^{\ell_2} \wedge \\
 & \hat{\mathbf{C}}(\ell_1) \subseteq \hat{\rho}(x) \wedge \hat{\mathbf{D}}(\ell_1) \sqsubseteq \hat{\delta}(x) \quad \wedge \quad \hat{\mathbf{C}}(\ell_2) \subseteq \hat{\mathbf{C}}(\ell) \wedge \hat{\mathbf{D}}(\ell_2) \sqsubseteq \hat{\mathbf{D}}(\ell)
 \end{aligned}$$

$$\begin{aligned}
 (\hat{\mathbf{C}}, \hat{\mathbf{D}}, \hat{\rho}, \hat{\delta}) \models_D (t_1^{\ell_1} \text{ op } t_2^{\ell_2})^\ell \\
 \text{iff } & (\hat{\mathbf{C}}, \hat{\mathbf{D}}, \hat{\rho}, \hat{\delta}) \models_D t_1^{\ell_1} \wedge (\hat{\mathbf{C}}, \hat{\mathbf{D}}, \hat{\rho}, \hat{\delta}) \models_D t_2^{\ell_2} \wedge \\
 & f_{op}(\hat{\mathbf{D}}(\ell_1), \hat{\mathbf{D}}(\ell_2)) \sqsubseteq \hat{\mathbf{D}}(\ell)
 \end{aligned}$$

Example:

	$(\hat{C}, \hat{\rho})$	$(\hat{C}, \hat{\rho})$	$(\hat{C}, \hat{\rho})$	$(\hat{D}, \hat{\delta})$
1	\emptyset	$\{+\}$	\emptyset	$\{+\}$
2	\emptyset	$\{0\}$	\emptyset	$\{0\}$
3	\emptyset	$\{tt\}$	\emptyset	$\{tt\}$
4	\emptyset	$\{0\}$	\emptyset	$\{0\}$
5	$\{\text{fn } y \Rightarrow y^4\}$	$\{\text{fn } y \Rightarrow y^4\}$	$\{\text{fn } y \Rightarrow y^4\}$	\emptyset
6	\emptyset	\emptyset	\emptyset	\emptyset
7	$\{\text{fn } z \Rightarrow 25^6\}$	\emptyset	\emptyset	\emptyset
8	$\{\text{fn } y \Rightarrow y^4, \text{fn } z \Rightarrow 25^6\}$	$\{\text{fn } y \Rightarrow y^4\}$	$\{\text{fn } y \Rightarrow y^4\}$	\emptyset
9	$\{\text{fn } x \Rightarrow (\dots)^8\}$	$\{\text{fn } x \Rightarrow (\dots)^8\}$	$\{\text{fn } x \Rightarrow (\dots)^8\}$	\emptyset
10	$\{\text{fn } x \Rightarrow (\dots)^8\}$	$\{\text{fn } x \Rightarrow (\dots)^8\}$	$\{\text{fn } x \Rightarrow (\dots)^8\}$	\emptyset
11	\emptyset	$\{+\}$	\emptyset	$\{+\}$
12	$\{\text{fn } y \Rightarrow y^4, \text{fn } z \Rightarrow 25^6\}$	$\{\text{fn } y \Rightarrow y^4\}$	$\{\text{fn } y \Rightarrow y^4\}$	\emptyset
13	\emptyset	$\{0\}$	\emptyset	$\{0\}$
14	\emptyset	$\{0\}$	\emptyset	$\{0\}$
15	\emptyset	$\{0\}$	\emptyset	$\{0\}$
f	$\{\text{fn } x \Rightarrow (\dots)^8\}$	$\{\text{fn } x \Rightarrow (\dots)^8\}$	$\{\text{fn } x \Rightarrow (\dots)^8\}$	\emptyset
x	\emptyset	$\{+\}$	\emptyset	$\{+\}$
y	\emptyset	$\{0\}$	\emptyset	$\{0\}$
z	\emptyset	\emptyset	\emptyset	\emptyset

Staging the specification

Alternative clause for the conditional where the data flow component *cannot* influence the control flow component:

$$\begin{aligned} (\hat{C}, \hat{D}, \hat{\rho}, \hat{\delta}) \models_D (\text{if } t_0^{\ell_0} \text{ then } t_1^{\ell_1} \text{ else } t_2^{\ell_2})^\ell \\ \text{iff} \quad & (\hat{C}, \hat{D}, \hat{\rho}, \hat{\delta}) \models_D t_0^{\ell_0} \wedge \\ & (\hat{C}, \hat{D}, \hat{\rho}, \hat{\delta}) \models_D t_1^{\ell_1} \wedge \hat{C}(\ell_1) \subseteq \hat{C}(\ell) \wedge \hat{D}(\ell_1) \sqsubseteq \hat{D}(\ell) \wedge \\ & (\hat{C}, \hat{D}, \hat{\rho}, \hat{\delta}) \models_D t_2^{\ell_2} \wedge \hat{C}(\ell_2) \subseteq \hat{C}(\ell) \wedge \hat{D}(\ell_2) \sqsubseteq \hat{D}(\ell) \end{aligned}$$

Compare with flow-insensitive Data Flow Analyses.

Adding Context Information

Mono-variant analysis: does not distinguish the various instances of variables and program points from one another. (Compare with context-insensitive interprocedural analysis.) 0-CFA is a typical example.

Poly-variant analysis: distinguishes between the various instances of variables and program points. (Compare with context-sensitive interprocedural analysis.)

Example:

$(\text{let } f = (\text{fn } x \Rightarrow x^1)^2 \text{ in } ((f^3 f^4)^5 (\text{fn } y \Rightarrow y^6)^7)^8)^9$

The least 0-CFA analysis:

$$\begin{array}{ll} \hat{C}_{id}(1) = \{\text{fn } x \Rightarrow x^1, \text{fn } y \Rightarrow y^6\} & \hat{C}_{id}(2) = \{\text{fn } x \Rightarrow x^1\} \\ \hat{C}_{id}(3) = \{\text{fn } x \Rightarrow x^1\} & \hat{C}_{id}(4) = \{\text{fn } x \Rightarrow x^1\} \\ \hat{C}_{id}(5) = \{\text{fn } x \Rightarrow x^1, \text{fn } y \Rightarrow y^6\} & \hat{C}_{id}(6) = \{\text{fn } y \Rightarrow y^6\} \\ \hat{C}_{id}(7) = \{\text{fn } y \Rightarrow y^6\} & \hat{C}_{id}(8) = \{\text{fn } x \Rightarrow x^1, \text{fn } y \Rightarrow y^6\} \\ \hat{C}_{id}(9) = \{\text{fn } x \Rightarrow x^1, \text{fn } y \Rightarrow y^6\} & \\ \hat{\rho}_{id}(f) = \{\text{fn } x \Rightarrow x^1\} & \hat{\rho}_{id}(x) = \{\text{fn } x \Rightarrow x^1, \text{fn } y \Rightarrow y^6\} \\ \hat{\rho}_{id}(y) = \{\text{fn } y \Rightarrow y^6\} & \end{array}$$

The analysis says that the expression may evaluate to $\text{fn } x \Rightarrow x^1$ or $\text{fn } y \Rightarrow y^6$.

However, only $\text{fn } y \Rightarrow y^6$ is a possible result.

A purely syntactic solution:

Expand

```
(let f = (fn x => x) in ((f f) (fn y => y)))
```

into

```
let f1 = (fn x1 => x1)
in let f2 = (fn x2 => x2) in (f1 f2) (fn y => y)
```

and analyse the expanded expression.

The 0-CFA analysis is now able to deduce that the overall expression will evaluate to `fn y => y` only.

A purely semantic solution: Uniform k -CFA

Idea: extend the set $\widehat{\text{Val}}$ to include context information

In a (uniform) k -CFA a context δ records the last k dynamic call points; hence contexts will be sequences of labels of length at most k and they will be updated whenever a function application is analysed. (Compare call strings of length at most k .)

Abstract Domains

$\delta \in \Delta = \text{Lab}^{\leq k}$ context information

$ce \in \text{CEnv} = \text{Var} \rightarrow \Delta$ context environments

$\hat{v} \in \widehat{\text{Val}} = \mathcal{P}(\text{Term} \times \text{CEnv})$ abstract values

$\hat{\rho} \in \widehat{\text{Env}} = (\text{Var} \times \Delta) \rightarrow \widehat{\text{Val}}$ abstract environments

$\hat{c} \in \widehat{\text{Cache}} = (\text{Lab} \times \Delta) \rightarrow \widehat{\text{Val}}$ abstract caches

(Uniform because Δ used both for $\widehat{\text{Env}}$ and $\widehat{\text{Cache}}$.)

Acceptability Relation

$$(\hat{C}, \hat{\rho}) \models_{\delta}^{ce} e$$

where

- ce is the current context environment – will be changed when new bindings are made
- δ is the current context – will be changed when functions are called

Idea: The formula expresses that $(\hat{C}, \hat{\rho})$ is an acceptable analysis of e in the *context* specified by ce and δ .

Control Flow Analysis with Context (1)

$$(\hat{C}, \hat{\rho}) \models_{\delta}^{ce} (\text{fn } x \Rightarrow e_0)^{\ell} \quad \text{iff} \quad \{(\text{fn } x \Rightarrow e_0, \text{ce})\} \subseteq \hat{C}(\ell, \delta)$$

$$(\hat{C}, \hat{\rho}) \models_{\delta}^{ce} (\text{fun } f \ x \Rightarrow e_0)^{\ell} \quad \text{iff} \quad \{(\text{fun } f \ x \Rightarrow e_0, \text{ce})\} \subseteq \hat{C}(\ell, \delta)$$

$$\begin{aligned} (\hat{C}, \hat{\rho}) \models_{\delta}^{ce} (t_1^{\ell_1} \ t_2^{\ell_2})^{\ell} \\ \text{iff} \quad & (\hat{C}, \hat{\rho}) \models_{\delta}^{ce} t_1^{\ell_1} \wedge (\hat{C}, \hat{\rho}) \models_{\delta}^{ce} t_2^{\ell_2} \wedge \\ & (\forall (\text{fn } x \Rightarrow t_0^{\ell_0}, \text{ce}_0) \in \hat{C}(\ell_1, \delta) : \\ & \quad (\hat{C}, \hat{\rho}) \models_{\delta_0}^{ce'_0} t_0^{\ell_0} \wedge \hat{C}(\ell_2, \delta) \subseteq \hat{\rho}(x, \delta_0) \wedge \hat{C}(\ell_0, \delta_0) \subseteq \hat{C}(\ell, \delta) \\ & \quad \text{where } \delta_0 = \lceil \delta, \ell \rceil_k \text{ and } ce'_0 = ce_0[x \mapsto \delta_0]) \wedge \\ & (\forall (\text{fun } f \ x \Rightarrow t_0^{\ell_0}, \text{ce}_0) \in \hat{C}(\ell_1, \delta) : \\ & \quad (\hat{C}, \hat{\rho}) \models_{\delta_0}^{ce'_0} t_0^{\ell_0} \wedge \hat{C}(\ell_2, \delta) \subseteq \hat{\rho}(f, \delta_0) \wedge \hat{C}(\ell_0, \delta_0) \subseteq \hat{C}(\ell, \delta) \wedge \\ & \quad \{(\text{fun } f \ x \Rightarrow t_0^{\ell_0}, \text{ce}_0)\} \subseteq \hat{\rho}(f, \delta_0) \\ & \quad \text{where } \delta_0 = \lceil \delta, \ell \rceil_k \text{ and } ce'_0 = ce_0[f \mapsto \delta_0, x \mapsto \delta_0]) \end{aligned}$$

Control Flow Analysis with Context (2)

$$(\hat{C}, \hat{\rho}) \models_{\delta}^{ce} c^{\ell} \text{ always}$$

$$(\hat{C}, \hat{\rho}) \models_{\delta}^{ce} x^{\ell} \quad \underline{\text{iff}} \quad \hat{\rho}(x, ce(x)) \subseteq \hat{C}(\ell, \delta)$$

$$\begin{aligned} (\hat{C}, \hat{\rho}) \models_{\delta}^{ce} (\text{if } t_0^{\ell_0} \text{ then } t_1^{\ell_1} \text{ else } t_2^{\ell_2})^{\ell} \\ \underline{\text{iff}} \quad (\hat{C}, \hat{\rho}) \models_{\delta}^{ce} t_0^{\ell_0} \wedge (\hat{C}, \hat{\rho}) \models_{\delta}^{ce} t_1^{\ell_1} \wedge (\hat{C}, \hat{\rho}) \models_{\delta}^{ce} t_2^{\ell_2} \wedge \\ \hat{C}(\ell_1, \delta) \subseteq \hat{C}(\ell, \delta) \wedge \hat{C}(\ell_2, \delta) \subseteq \hat{C}(\ell, \delta) \end{aligned}$$

$$\begin{aligned} (\hat{C}, \hat{\rho}) \models_{\delta}^{ce} (\text{let } x = t_1^{\ell_1} \text{ in } t_2^{\ell_2})^{\ell} \\ \underline{\text{iff}} \quad (\hat{C}, \hat{\rho}) \models_{\delta}^{ce} t_1^{\ell_1} \wedge (\hat{C}, \hat{\rho}) \models_{\delta}^{ce'} t_2^{\ell_2} \wedge \\ \hat{C}(\ell_1, \delta) \subseteq \hat{\rho}(x, \delta) \wedge \hat{C}(\ell_2, \delta) \subseteq \hat{C}(\ell, \delta) \\ \text{where } ce' = ce[x \mapsto \delta] \end{aligned}$$

$$(\hat{C}, \hat{\rho}) \models_{\delta}^{ce} (t_1^{\ell_1} \text{ op } t_2^{\ell_2})^{\ell} \quad \underline{\text{iff}} \quad (\hat{C}, \hat{\rho}) \models_{\delta}^{ce} t_1^{\ell_1} \wedge (\hat{C}, \hat{\rho}) \models_{\delta}^{ce} t_2^{\ell_2}$$

Example:

$(\text{let } f = (\text{fn } x \Rightarrow x^1)^2 \text{ in } ((f^3 f^4)^5 (\text{fn } y \Rightarrow y^6)^7)^8)^9$

Contexts of interest for uniform 1-CFA:

- Λ : the initial context
- 5: the context when the application point labelled 5 has been passed
- 8: the context when the application point labelled 8 has been passed

Context environments of interest for uniform 1-CFA:

- $ce_0 = []$ the initial (empty) context environment
- $ce_1 = ce_0[f \mapsto \Lambda]$ the context environment for the analysis of the body of the `let`-construct
- $ce_2 = ce_0[x \mapsto 5]$ the context environment used for the analysis of the body of f initiated at the application point 5
- $ce_3 = ce_0[x \mapsto 8]$ the context environment used for the analysis of the body of f initiated at the application point 8.

Example: Let us take \hat{C}_{id}' and $\hat{\rho}_{id}'$ to be:

$$\begin{aligned}
\hat{C}_{id}'(1, 5) &= \{(\text{fn } x \Rightarrow x^1, \text{ce}_0)\} & \hat{C}_{id}'(1, 8) &= \{(\text{fn } y \Rightarrow y^6, \text{ce}_0)\} \\
\hat{C}_{id}'(2, \wedge) &= \{(\text{fn } x \Rightarrow x^1, \text{ce}_0)\} & \hat{C}_{id}'(3, \wedge) &= \{(\text{fn } x \Rightarrow x^1, \text{ce}_0)\} \\
\hat{C}_{id}'(4, \wedge) &= \{(\text{fn } x \Rightarrow x^1, \text{ce}_0)\} & \hat{C}_{id}'(5, \wedge) &= \{(\text{fn } x \Rightarrow x^1, \text{ce}_0)\} \\
\hat{C}_{id}'(7, \wedge) &= \{(\text{fn } y \Rightarrow y^6, \text{ce}_0)\} & \hat{C}_{id}'(8, \wedge) &= \{(\text{fn } y \Rightarrow y^6, \text{ce}_0)\} \\
\hat{C}_{id}'(9, \wedge) &= \{(\text{fn } y \Rightarrow y^6, \text{ce}_0)\} & & \\
\hat{\rho}_{id}'(f, \wedge) &= \{(\text{fn } x \Rightarrow x^1, \text{ce}_0)\} & & \\
\hat{\rho}_{id}'(x, 5) &= \{(\text{fn } x \Rightarrow x^1, \text{ce}_0)\} & \hat{\rho}_{id}'(x, 8) &= \{(\text{fn } y \Rightarrow y^6, \text{ce}_0)\}
\end{aligned}$$

This is an acceptable analysis result:

$$(\hat{C}_{id}', \hat{\rho}_{id}') \models_{\wedge}^{\text{ce}_0} (\text{let } f = (\text{fn } x \Rightarrow x^1)^2 \text{ in } ((f^3 f^4)^5 (\text{fn } y \Rightarrow y^6)^7)^8)^9$$

Complexity

Uniform k -CFA has exponential worst case complexity even when $k = 1$

Assume that the expression has size n and that it has p different variables. Then Δ has $O(n)$ elements and hence there will be $O(p \cdot n)$ different pairs (x, δ) and $O(n^2)$ different pairs (ℓ, δ) . This means that $(\hat{C}, \hat{\rho})$ can be seen as an $O(n^2)$ tuple of values from $\widehat{\text{Val}}$. Since $\widehat{\text{Val}}$ itself is a powerset of pairs of the form (t, ce) and there are $O(n \cdot n^p)$ such pairs it follows that $\widehat{\text{Val}}$ has height $O(n \cdot n^p)$. Since $O(p) = O(n)$ we have the exponential worst case complexity.

0-CFA analysis has polynomial worst case complexity

It corresponds to letting Δ be a singleton. Repeating the above calculations we can see $(\hat{C}, \hat{\rho})$ as an $O(p + n)$ tuple of values from $\widehat{\text{Val}}$, and $\widehat{\text{Val}}$ will be a lattice of height $O(n)$.

Variations (based on call-strings)

Uniform k -CFA

$$\begin{array}{llll} ce \in \mathbf{CEnv} & = & \mathbf{Var} \rightarrow \Delta & \text{context environments} \\ \hat{v} \in \widehat{\mathbf{Val}} & = & \mathcal{P}(\mathbf{Term} \times \mathbf{CEnv}) & \text{abstract values} \\ \hat{\rho} \in \widehat{\mathbf{Env}} & = & (\mathbf{Var} \times \Delta) \rightarrow \widehat{\mathbf{Val}} & \text{abstract environments} \\ \hat{\mathbf{C}} \in \widehat{\mathbf{Cache}} & = & (\mathbf{Lab} \times \Delta) \rightarrow \widehat{\mathbf{Val}} & \text{abstract caches} \end{array}$$

k -CFA

$$\hat{\mathbf{C}} \in \widehat{\mathbf{Cache}} = (\mathbf{Lab} \times \mathbf{CEnv}) \rightarrow \widehat{\mathbf{Val}} \quad \text{abstract caches}$$

Polynomial k -CFA

$$\hat{v} \in \widehat{\mathbf{Val}} = \mathcal{P}(\mathbf{Term} \times \Delta) \quad \text{abstract values}$$