Principles of Abstract Interpretation
MIT press

Ch. **3**, Syntax, semantics, properties, and static analysis of expressions

Patrick Cousot

pcousot.github.io

PrAbsInt@gmail.com github.com/PrAbsInt/

These slides are available at
http://github.com/PrAbsInt/slides/slides/slides-03--expressions-PrAbsInt.pdf

Chapter 3

Ch. **3**, Syntax, semantics, properties, and static analysis of expressions

The objective of this Chapter **3** (Syntax, semantics, properties, and static analysis of expressions) is to introduce abstract interpretation using an extremely simple example: the rule of signs

Product of two integers

$\cdot$	$-$	$+$
$-$	$+$	$-$
$+$	$-$	$+$

In words, we have:

- Minus times Minus gives Plus
- Minus times Plus gives Minus
- Plus times Minus gives Minus
- Plus times Plus gives Plus

en.wikipedia.org/wiki/Product_(mathematics)

Brahmagupta

- Brahmagupta (born c. 598 CE[1], died after 665 CE) was an Indian mathematician and astronomer;
- Invented the rule of signs (including to compute with zero);
- Probably the very first recorded historical example of abstract interpretation :)

en.wikipedia.org/wiki/Brahmagupta

[1] Common Era

Syntax of expressions

Syntax of expressions

$x, y, \ldots$	$\in$	$\mathbb{V}$			variables ($\mathbb{V}$ not empty)
A	$\in$	$\mathbb{A}$	$::=$	$1 \mid x \mid A_1 - A_2$	arithmetic expressions
B	$\in$	$\mathbb{B}$	$::=$	$A_1 < A_2 \mid B_1 \text{ nand } B_2$	boolean expressions
E	$\in$	$\mathbb{E}$	$::=$	$A \mid B$	expressions

This is an example of *context-free grammar*.

Binary operators are left associative and arithmetic operators have priority over boolean operators (so $1 - 1 < 1 - 1 - 1$ is $((1 - 1) < ((1 - 1) - 1))$ *i.e.* false ff).

en.wikipedia.org/wiki/Syntax_(programming_languages)
en.wikipedia.org/wiki/Context-free_grammar

Semantics of expressions

Environment

- The value of an expression depends on the value of the free variables *e.g.*
  $$\texttt{x} - 1 \text{ is } 2 \text{ when } \texttt{x} = 3,\ \texttt{x} - 1 \text{ is } 42 \text{ when } \texttt{x} = 43,\ \textit{etc.};$$
- We cannot enumerate the infinitely many cases;
- The computer uses values of variables stored in memory;
- The evaluation of expressions by the computer can be explained independently of the memory content;
- We formalize the memory by environments assigning values to variables (assignments in logic);
- An environment
  $$\rho \in \mathbb{V} \to \mathbb{Z}$$
  is a total function ρ mapping a variable $\texttt{x} \in \mathbb{V}$ to its integer value $\rho(\texttt{x}) \in \mathbb{Z}$;

en.wikipedia.org/wiki/Typing_environment
en.wikipedia.org/wiki/Valuation_(logic)

Semantics of expressions

$$
\begin{array}{rcll}
\mathscr{A}\llbracket 1 \rrbracket\rho & \triangleq & 1 & \\
\mathscr{A}\llbracket \mathtt{x} \rrbracket\rho & \triangleq & \rho(\mathtt{x}) & \\
\mathscr{A}\llbracket \mathtt{A}_1 - \mathtt{A}_2 \rrbracket\rho & \triangleq & \mathscr{A}\llbracket \mathtt{A}_1 \rrbracket\rho - \mathscr{A}\llbracket \mathtt{A}_2 \rrbracket\rho & \\
\mathscr{B}\llbracket \mathtt{A}_1 < \mathtt{A}_2 \rrbracket\rho & \triangleq & \mathscr{A}\llbracket \mathtt{A}_1 \rrbracket\rho < \mathscr{A}\llbracket \mathtt{A}_2 \rrbracket\rho & \\
\mathscr{B}\llbracket \mathtt{B}_1 \text{ nand } \mathtt{B}_2 \rrbracket\rho & \triangleq & \mathscr{B}\llbracket \mathtt{B}_1 \rrbracket\rho \uparrow \mathscr{B}\llbracket \mathtt{B}_2 \rrbracket\rho & \\
\mathscr{S}\llbracket \mathtt{E} \rrbracket & \triangleq & \mathscr{A}\llbracket \mathtt{E} \rrbracket & \text{when} \quad \mathtt{E} \in \mathbb{A} \\
\mathscr{S}\llbracket \mathtt{E} \rrbracket & \triangleq & \mathscr{B}\llbracket \mathtt{E} \rrbracket & \text{when} \quad \mathtt{E} \in \mathbb{B}
\end{array} \tag{3.4}
$$

a	tt	tt	ff	ff
b	tt	ff	tt	ff
$a \uparrow b$	ff	tt	tt	tt

- This is an example of well-defined structural definition.
- $\mathscr{A}\llbracket \mathtt{A} \rrbracket$ and $\mathscr{B}\llbracket \mathtt{B} \rrbracket$ are total functions (in $\mathbb{Z}$), proof by structural induction.

`en.wikipedia.org/wiki/Semantics_(computer_science)`

Semantic properties of expressions

Properties

- We represent a property by the set of elements that have this property.
- For example
  - "x is an even natural" is "$x \in \{0, 2, 4, \ldots\}$".
  - "x is constant equal to 1" is "$x \in \{1\}$".

  So a property of a natural is an element of $\wp(\mathbb{N})$.

  For example
  - The property $\{0, 2, 4, \ldots\}$ is "to be even".
  - The property $\{1\}$ is "to be one".

Powerset

- If S is a set then $\wp(S)$ is the *powerset* of S,

$$\wp(S) \quad \triangleq \quad \{X \mid X \subseteq S\}$$

- Example: $\wp(\{0,1\}) \quad \triangleq \quad \{\emptyset, \{0\}, \{1\}, \{0,1\}\}$
- Hasse diagram:

$$\wp(\{0,1\}) \quad \triangleq \quad \begin{tikzcd}[column sep=small] & \{0,1\} & \\ \{0\} \arrow[ur, dash] & & \{1\} \arrow[ul, dash] \\ & \emptyset \arrow[ul, dash] \arrow[ur, dash] & \end{tikzcd}$$

en.wikipedia.org/wiki/Power_set
en.wikipedia.org/wiki/Hasse_diagram

Implication, weaker and stronger properties

- When considering properties as sets, logical implication is subset inclusion $\subseteq$.
- For example "to be greater that 42 implies to be positive" is $\{x \in \mathbb{Z} \mid x > 42\} \subseteq \{x \in \mathbb{Z} \mid x \geqslant 0\}$.
- If $P \subseteq Q$ then P is said to be stronger/more precise than Q and Q is said to be weaker/less precise that P.
- Stronger/more precise properties are satisfied by less elements while weaker/less precise properties are satisfied by more elements.
- False ff *i.e.* $\emptyset$ is the strongest property while true tt *i.e.* $\mathbb{Z}$ is the weakest property of integers.
- conjunction $\wedge$ is intersection $\cap$ and disjunction $\vee$ is union $\cup$.

en.wikipedia.org/wiki/Logical_consequence
en.wikipedia.org/wiki/Subset

Semantics properties of expressions

- By property of an expression, we mean a semantic property, that is a property of its semantics;
- The semantic belongs to $(\mathbb{V} \to \mathbb{Z}) \to \mathbb{Z}$;
- So a semantic property is an element of $\wp((\mathbb{V} \to \mathbb{Z}) \to \mathbb{Z})$;
- Arithmetic expression $\mathtt{A}$ is said to have semantic property $P \in \wp((\mathbb{V} \to \mathbb{Z}) \to \mathbb{Z})$ if and only if $\mathscr{A}[\![\mathtt{A}]\!] \in P$;
- Semantic properties P of expressions are just a particular case of property of expressions *i.e.* the property $\{\mathtt{A} \in \mathbb{E} \mid \mathscr{A}[\![\mathtt{A}]\!] \in P\}$[2].

[2] This will be discussed in greater details in Chapter **9** (Undecidability and Rice theorem)

Collecting semantics of expressions

Collecting semantics of expressions

- The collecting semantics of expressions is the strongest property of an expression.

$$\mathcal{S}^{\mathbb{C}}[\![\mathsf{A}]\!] \triangleq \{\mathcal{A}[\![\mathsf{A}]\!]\} \in \wp((\mathbb{V} \to \mathbb{Z}) \to \mathbb{Z}) \tag{3.13}$$

- Arithmetic expression A is said to have semantic property $P \in \wp((\mathbb{V} \to \mathbb{Z}) \to \mathbb{Z})$ if and only if $\mathcal{A}[\![\mathsf{A}]\!] \in P$
- Equivalently $\mathcal{S}^{\mathbb{C}}[\![\mathsf{A}]\!] \subseteq P$ (so we don't need to use $\in$)
- $\mathcal{S}^{\mathbb{C}}[\![\mathsf{A}]\!]$ is the strongest property of A.

- The collecting semantics of boolean expressions is

$$\mathcal{S}^{\mathbb{C}}[\![\mathsf{B}]\!] \triangleq \{\mathcal{B}[\![\mathsf{B}]\!]\} \in \wp((\mathbb{V} \to \mathbb{Z}) \to \mathbb{B})$$

Structural collecting semantics

$$
\begin{aligned}
\mathcal{S}^{\mathbb{C}}[\![\mathtt{1}]\!] &= \{\rho \in (\mathbb{V} \to \mathbb{Z}) \mapsto 1\} \\
\mathcal{S}^{\mathbb{C}}[\![\mathtt{x}]\!] &= \{\rho \in (\mathbb{V} \to \mathbb{Z}) \mapsto \rho(\mathtt{x})\} \\
\mathcal{S}^{\mathbb{C}}[\![\mathtt{A}_1 \text{ - } \mathtt{A}_2]\!] &= \{\rho \in (\mathbb{V} \to \mathbb{Z}) \mapsto f_1(\rho) - f_2(\rho) \mid f_1 \in \mathcal{S}^{\mathbb{C}}[\![\mathtt{A}_1]\!] \wedge f_2 \in \mathcal{S}^{\mathbb{C}}[\![\mathtt{A}_2]\!]\}
\end{aligned}
$$

$$
\begin{aligned}
\mathcal{S}^{\mathbb{C}}[\![\mathtt{A}_1 < \mathtt{A}_2]\!] &= \{\rho \in (\mathbb{V} \to \mathbb{Z}) \mapsto f_1(\rho) < f_2(\rho) \mid f_1 \in \mathcal{S}^{\mathbb{C}}[\![\mathtt{A}_1]\!] \wedge f_2 \in \mathcal{S}^{\mathbb{C}}[\![\mathtt{A}_2]\!]\} \\
\mathcal{S}^{\mathbb{C}}[\![\mathtt{B}_1 \text{ nand } \mathtt{B}_2]\!] &= \{\rho \in (\mathbb{V} \to \mathbb{Z}) \mapsto f_1(\rho) \uparrow f_2(\rho) \mid f_1 \in \mathcal{S}^{\mathbb{C}}[\![\mathtt{B}_1]\!] \wedge f_2 \in \mathcal{S}^{\mathbb{C}}[\![\mathtt{B}_2]\!]\}
\end{aligned}
$$

$x \mapsto t$ is the function f such that for parameter x, the value $f(x)$ of f at x is equal to the value of the term t (depending upon x). $x \in X \mapsto t$ states that f is undefined when $x \notin X$.

Sign abstraction

Sign property (of an individual variable)

$$\mathbb{P}^{\pm} = \begin{tikzcd}[arrows=-]
 & \mathbb{Z} \arrow[dl] \arrow[d] \arrow[dr] & \\
\{z \mid z \leqslant 0\} \arrow[d] \arrow[dr] & \{z \mid z \neq 0\} \arrow[dl] \arrow[dr] & \{z \mid z \geqslant 0\} \arrow[dl] \arrow[d] \\
\{z \mid z < 0\} \arrow[dr] & \{0\} \arrow[d] & \{z \mid z > 0\} \arrow[dl] \\
 & \varnothing &
\end{tikzcd}$$

The Hasse diagram for partial order $\subseteq$, $\cup$ is the join, $\cap$ is the meet, *etc*.

Encoding of sign properties (of an individual variable)

$$\mathbb{P}^{\pm} = \begin{tikzcd}[column sep=small]
 & \top_{\pm} \arrow[dl, dash] \arrow[d, dash] \arrow[dr, dash] & \\
\leqslant 0 \arrow[d, dash] \arrow[dr, dash] & \neq 0 \arrow[dl, dash] \arrow[dr, dash] & \geqslant 0 \arrow[dl, dash] \arrow[d, dash] \\
<0 \arrow[dr, dash] & =0 \arrow[d, dash] & >0 \arrow[dl, dash] \\
 & \bot_{\pm} &
\end{tikzcd}$$

Concretization function:

$$\begin{array}{lcl@{\qquad}lcl}
\gamma_{\pm}(\bot_{\pm}) & \triangleq & \emptyset & \gamma_{\pm}(\leqslant 0) & \triangleq & \{z \mid z \leqslant 0\} \\
\gamma_{\pm}(<0) & \triangleq & \{z \mid z < 0\} & \gamma_{\pm}(\neq 0) & \triangleq & \{z \mid z \neq 0\} \\
\gamma_{\pm}(=0) & \triangleq & \{0\} & \gamma_{\pm}(\geqslant 0) & \triangleq & \{z \mid z \geqslant 0\} \\
\gamma_{\pm}(>0) & \triangleq & \{z \mid z > 0\} & \gamma_{\pm}(\top_{\pm}) & \triangleq & \mathbb{Z}
\end{array}$$

The lattice of abstract properties

$$\mathbb{P}^{\pm} = \begin{tikzcd}
 & \top_{\pm} \arrow[dl, dash] \arrow[d, dash] \arrow[dr, dash] & \\
\leqslant 0 \arrow[d, dash] \arrow[dr, dash] & \neq 0 \arrow[dl, dash] \arrow[dr, dash] & \geqslant 0 \arrow[dl, dash] \arrow[d, dash] \\
<0 \arrow[dr, dash] & =0 \arrow[d, dash] & >0 \arrow[dl, dash] \\
 & \bot_{\pm} &
\end{tikzcd}$$

The Hasse diagram for partial order $\sqsubseteq$, $\sqcup$ is the join, $\sqcap$ is the meet, *etc*.

e.g. $\bigsqcap\{\leqslant 0, \neq 0\} = <0, \quad \bigsqcap \varnothing = \top_{\pm}$

Encoding of sign properties (of an individual variable)

$$
\mathbb{P}^{\pm} = \quad
\begin{tikzcd}[column sep=small]
 & \top_{\pm} \arrow[dl, dash] \arrow[d, dash] \arrow[dr, dash] & \\
\leqslant 0 \arrow[d, dash] \arrow[dr, dash] & \neq 0 \arrow[dl, dash] \arrow[dr, dash] & \geqslant 0 \arrow[dl, dash] \arrow[d, dash] \\
<0 \arrow[dr, dash] & =0 \arrow[d, dash] & >0 \arrow[dl, dash] \\
 & \bot_{\pm} &
\end{tikzcd}
$$

Abstraction function:
$$
\begin{aligned}
\alpha_{\pm}(P) \;\triangleq\; & \llparenthesis\, P \subseteq \varnothing \;?\; \bot_{\pm} \\
& \parallel P \subseteq \{z \mid z < 0\} \;?\; {<}0 \\
& \parallel P \subseteq \{0\} \;?\; {=}0 \\
& \parallel P \subseteq \{z \mid z > 0\} \;?\; {>}0 \\
& \parallel P \subseteq \{z \mid z \leqslant 0\} \;?\; {\leqslant}0 \\
& \parallel P \subseteq \{z \mid z \neq 0\} \;?\; {\neq}0 \\
& \parallel P \subseteq \{z \mid z \geqslant 0\} \;?\; {\geqslant}0 \\
& \,\mathrm{:}\; \top_{\pm} \,\rrparenthesis
\end{aligned}
\tag{3.30}
$$

Galois connection

- The pair $\langle \alpha_\pm,\ \gamma_\pm \rangle$ of functions satisfies $\alpha_\pm(P) \sqsubseteq Q \Leftrightarrow P \subseteq \gamma_\pm(Q)$
- For example,

$$\big(\alpha_\pm(\{-2,-1\}) \quad \triangleq \quad {<}0 \quad \sqsubseteq \quad {\neq}0\big) \quad \Leftrightarrow \quad \big(\{-2,-1\} \quad \subseteq \quad \{z \mid z \neq 0\} \quad \triangleq \quad \gamma_\pm({\neq}0)\big)$$

- Let us prove that we have a Galois connection between concrete and abstract properties

Galois connection

- The pair $\langle\alpha_\pm,\ \gamma_\pm\rangle$ of functions satisfies $\alpha_\pm(P) \sqsubseteq Q \Leftrightarrow P \subseteq \gamma_\pm(Q)$

$$
\begin{array}{ll@{\qquad}r}
 & \alpha_\pm(P) \sqsubseteq Q & \\
\Leftrightarrow & \alpha_\pm(P) \sqsubseteq {\neq}0 & \text{(in case } Q = {\neq}0\text{, other cases are similar)} \\
\Leftrightarrow & \alpha_\pm(P) \in \{\bot_\pm, {<}0, {\neq}0, {>}0\} & \text{(def. } \sqsubseteq\text{)} \\
\Leftrightarrow & P \subseteq \emptyset \vee P \subseteq \{z \mid z < 0\} \vee P \subseteq \{z \mid z > 0\} \vee P \subseteq \{z \mid z \neq 0\} & \text{(def. } \alpha_\pm\text{)} \\
\Leftrightarrow & P \subseteq \{z \mid z \neq 0\} & \text{(def. } \subseteq\text{)} \\
\Leftrightarrow & P \subseteq \gamma_\pm({\neq}0) & \text{(def. } \gamma_\pm\text{)} \\
\Leftrightarrow & P \subseteq \gamma_\pm(Q) & \text{(case } Q = {\neq}0\text{)}
\end{array}
$$

- This is the definition of a Galois connection
- We write $\langle\wp(\mathbb{Z}),\ \subseteq\rangle \underset{\alpha_\pm}{\overset{\gamma_\pm}{\leftrightarrows}} \langle\mathbb{P}^\pm,\ \sqsubseteq\rangle$
- This will be further generalized.

en.wikipedia.org/wiki/Galois_connection
en.wikipedia.org/wiki/Évariste_Galois

Sign abstract semantics

$$\begin{array}{rcl}\mathcal{S}[\![\mathtt{A}]\!] & \in & (\mathbb{V} \rightarrow \mathbb{P}^{\pm}) \rightarrow \mathbb{P}^{\pm} \\ \mathcal{S}[\![\mathtt{1}]\!]P & \triangleq & >0 \\ \mathcal{S}[\![\mathtt{x}]\!]P & \triangleq & P(\mathtt{x}) \\ \mathcal{S}[\![\mathtt{A}_1 - \mathtt{A}_2]\!]P & \triangleq & \mathcal{S}[\![\mathtt{A}_1]\!]P -_{\pm} \mathcal{S}[\![\mathtt{A}_2]\!]P\end{array} \tag{3.21}$$

	$x -_{\pm} y$	y							
		$\perp_{\pm}$	<0	$=0$	>0	$\leqslant 0$	$\neq 0$	$\geqslant 0$	$\top_{\pm}$
	$\perp_{\pm}$	$\perp_{\pm}$	$\perp_{\pm}$	$\perp_{\pm}$	$\perp_{\pm}$	$\perp_{\pm}$	$\perp_{\pm}$	$\perp_{\pm}$	$\perp_{\pm}$
	<0	$\perp_{\pm}$	$\top_{\pm}$	<0	<0	$\top_{\pm}$	$\top_{\pm}$	<0	$\top_{\pm}$
	$=0$	$\perp_{\pm}$	>0	$=0$	<0	$\geqslant 0$	$\neq 0$	$\leqslant 0$	$\top_{\pm}$
x	>0	$\perp_{\pm}$	>0	>0	$\top_{\pm}$	>0	$\top_{\pm}$	$\top_{\pm}$	$\top_{\pm}$
	$\leqslant 0$	$\perp_{\pm}$	>0	$\leqslant 0$	$\top_{\pm}$	$\top_{\pm}$	$\top_{\pm}$	$\leqslant 0$	$\top_{\pm}$
	$\neq 0$	$\perp_{\pm}$	$\top_{\pm}$	$\neq 0$	$\top_{\pm}$	$\top_{\pm}$	$\top_{\pm}$	$\top_{\pm}$	$\top_{\pm}$
	$\geqslant 0$	$\perp_{\pm}$	>0	$\geqslant 0$	$\top_{\pm}$	$\geqslant 0$	$\top_{\pm}$	$\top_{\pm}$	$\top_{\pm}$
	$\top_{\pm}$	$\perp_{\pm}$	$\top_{\pm}$	$\top_{\pm}$	$\top_{\pm}$	$\top_{\pm}$	$\top_{\pm}$	$\top_{\pm}$	$\top_{\pm}$

This is a specification of an abstract interpreter I

```
type aexpr = One |Var of string | Minus of aexpr * aexpr;;

let bot = 0 and neg = 1 and is0 = 2 and pos = 3 and
    neg0 = 4 and not0 = 5 and pos0 = 6 and top = 7;;

let print s = match s with
     0 -> "bot" | 1 -> "neg" | 2 -> "is0" | 3 -> "pos" |
     4 -> "neg0" | 5 ->  "not0" | 6 -> "pos0" | 7 -> "top" |
     _ -> failwith "incorrect sign";;

let minus= [|[|bot;  bot;  bot;  bot;  bot;  bot;  bot;  bot|];
             [|bot;  top;  neg;  neg;  top;  top;  neg;  top|];
             [|bot;  pos;  is0;  neg;  pos0; not0; neg0; top|];
             [|bot;  pos;  pos;  top;  pos;  top;  top;  top|];
             [|bot;  pos;  neg0; top;  top;  top;  neg0; top|];
             [|bot;  top;  not0; top;  top;  top;  top;  top|];
             [|bot;  pos;  pos0; top;  pos0; top;  top;  top|];
             [|bot;  top;  top;  top;  top;  top;  top;  top|];
           |];;
```

This is a specification of an abstract interpreter II

```
type environment = (string * int) list;;

let rec sign a r = match a with
  | One -> pos
  | Var x -> List.assoc x r
  | Minus (a1, a2) -> minus.(sign a1 r).(sign a2 r);;

- : aexpr -> environment -> int = <fun>

let r = [("x",pos); ("y",neg)];;

print (sign (Minus ((Var "x"),(Var "y"))) r);;

- : string = "pos"
```

Calculational design of the rule of signs

$$
\begin{array}{rl}
& >0 \mathbin{-_{\pm}} \leqslant 0 \\
\triangleq & \alpha_{\pm}(\{x - y \mid x \in \gamma_{\pm}(>0) \wedge y \in \gamma_{\pm}(\leqslant 0)\}) \\
= & \alpha_{\pm}(\{x - y \mid x > 0 \wedge y \leqslant 0\}) \\
= & \alpha_{\pm}(\{z \mid z > 0\}) \\
& \qquad \wr \text{for } \subseteq,\ x > 0 \wedge y \leqslant 0 \Rightarrow x - y > 0; \\
& \qquad \ \ \text{for } \supseteq \text{ if } z > 0 \text{ then take } x = z \text{ and } y = 0 \text{ so } z \in \{x - y \mid x > 0 \wedge -y \geqslant 0\}\ \wr \\
= & >0
\end{array}
$$

Same calculus for all other cases (can be automated with a theorem prover).

Soundness

Sign concretization

- Sign

$$
\begin{array}{lcllcl}
\gamma_\pm(\bot_\pm) & \triangleq & \emptyset & \gamma_\pm(\leqslant 0) & \triangleq & \{z \in \mathbb{Z} \mid z \leqslant 0\} \\
\gamma_\pm(<0) & \triangleq & \{z \in \mathbb{Z} \mid z < 0\} & \gamma_\pm(\neq 0) & \triangleq & \{z \in \mathbb{Z} \mid z \neq 0\} \\
\gamma_\pm(=0) & \triangleq & \{0\} & \gamma_\pm(\geqslant 0) & \triangleq & \{z \in \mathbb{Z} \mid z \geqslant 0\} \\
\gamma_\pm(>0) & \triangleq & \{z \in \mathbb{Z} \mid z > 0\} & \gamma_\pm(\top_\pm) & \triangleq & \mathbb{Z}
\end{array}
\tag{3.23}
$$

- Sign environment

$$
\dot{\gamma}_\pm(\overset{\pm}{\rho}) \triangleq \{\rho \in \mathbb{V} \to \mathbb{Z} \mid \forall \mathsf{x} \in \mathbb{V} \,.\, \rho(\mathsf{x}) \in \gamma_\pm(\overset{\pm}{\rho}(\mathsf{x}))\}
\tag{3.24}
$$

- Sign abstract property

$$
\ddot{\gamma}_\pm(\overline{P}) \triangleq \{\mathcal{S} \in (\mathbb{V} \to \mathbb{Z}) \to \mathbb{Z} \mid \forall \overset{\pm}{\rho} \in \mathbb{V} \to \mathbb{P}^\pm \,.\, \forall \rho \in \dot{\gamma}_\pm(\overset{\pm}{\rho}) \,.\, \mathcal{S}(\rho) \in \gamma_\pm(\overline{P}(\overset{\pm}{\rho}))\}
\tag{3.25}
$$

Sign abstraction

- Value property

$$
\begin{aligned}
\alpha_{\pm}(P) \quad \triangleq \quad & \llparenthesis\, P \subseteq \varnothing \;?\; \bot_{\pm} \\
& \,\|\, P \subseteq \{z \mid z < 0\} \;?\; {<}0 \\
& \,\|\, P \subseteq \{0\} \;?\; {=}0 \\
& \,\|\, P \subseteq \{z \mid z > 0\} \;?\; {>}0 \\
& \,\|\, P \subseteq \{z \mid z \leqslant 0\} \;?\; {\leqslant}0 \\
& \,\|\, P \subseteq \{z \mid z \neq 0\} \;?\; {\neq}0 \\
& \,\|\, P \subseteq \{z \mid z \geqslant 0\} \;?\; {\geqslant}0 \\
& \,:\, \top_{\pm} \,\rrparenthesis
\end{aligned}
\tag{3.30}
$$

- Environment property

$$
\dot{\alpha}_{\pm}(P) \quad \triangleq \quad \mathsf{x} \in \mathbb{V} \mapsto \alpha_{\pm}(\{\rho(\mathsf{x}) \mid \rho \in P\})
\tag{3.33}
$$

- Semantics property

$$
\ddot{\alpha}_{\pm}(P) \quad \triangleq \quad \overset{\pm}{\rho} \in \mathbb{V} \to \mathbb{P}^{\pm} \mapsto \alpha_{\pm}(\{\mathcal{S}(\rho) \mid \mathcal{S} \in P \wedge \rho \in \dot{\gamma}_{\pm}(\overset{\pm}{\rho})\})
\tag{3.34}
$$

Example of environment property abstraction

- The property of environments such that x is equal to 1:

$$\{\rho \in \mathbb{V} \to \mathbb{Z} \mid \rho(\mathsf{x}) = 1\}$$

- Sign abstraction:

$$\begin{aligned}
&\dot{\alpha}_{\pm}(\{\rho \in \mathbb{V} \to \mathbb{Z} \mid \rho(\mathsf{x}) = 1\})\\
\triangleq\ & \mathsf{y} \in \mathbb{V} \mapsto \alpha_{\pm}(\{\rho(\mathsf{y}) \mid \rho \in \{\rho \in \mathbb{V} \to \mathbb{Z} \mid \rho(\mathsf{x}) = 1\}\})\\
=\ & \mathsf{y} \in \mathbb{V} \mapsto \llparenthesis\, \mathsf{y} = \mathsf{x} \mathrel{?} \alpha_{\pm}(\{1\}) \mathrel{:} \alpha_{\pm}(\mathbb{Z}) \,\rrparenthesis\\
=\ & \mathsf{y} \in \mathbb{V} \mapsto \llparenthesis\, \mathsf{y} = \mathsf{x} \mathrel{?} {>}0 \mathrel{:} \top_{\pm} \,\rrparenthesis
\end{aligned}$$

- Sign concretization:

$$\begin{aligned}
&\dot{\gamma}_{\pm}(\mathsf{y} \in \mathbb{V} \mapsto \llparenthesis\, \mathsf{y} = \mathsf{x} \mathrel{?} {>}0 \mathrel{:} \top_{\pm} \,\rrparenthesis)\\
\triangleq\ & \{\rho \in \mathbb{V} \to \mathbb{Z} \mid \forall \mathsf{z} \in \mathbb{V} \,.\, \rho(\mathsf{z}) \in \gamma_{\pm}(\mathsf{y} \in \mathbb{V} \mapsto \llparenthesis\, \mathsf{y} = \mathsf{x} \mathrel{?} {>}0 \mathrel{:} \top_{\pm} \,\rrparenthesis(\mathsf{z}))\}\\
=\ & \{\rho \in \mathbb{V} \to \mathbb{Z} \mid \rho(\mathsf{x}) > 0\}
\end{aligned}$$

Galois connections

- Value to sign

$$\langle \wp(\mathbb{Z}),\ \subseteq\rangle \xleftarrow[\alpha_\pm]{\gamma_\pm} \langle \mathbb{P}^\pm,\ \sqsubseteq\rangle$$

- Value environment to sign environment

$$\langle \wp(\mathbb{V} \rightarrow \mathbb{Z}),\ \subseteq\rangle \xleftarrow[\dot{\alpha}_\pm]{\dot{\gamma}_\pm} \langle \mathbb{V} \rightarrow \mathbb{P}^\pm,\ \dot{\sqsubseteq}_\pm\rangle$$

- Semantic to sign abstract semantic property

$$\langle \wp((\mathbb{V} \rightarrow \mathbb{Z}) \rightarrow \mathbb{Z}),\ \subseteq\rangle \xleftarrow[\ddot{\alpha}_\pm]{\ddot{\gamma}_\pm} \langle (\mathbb{V} \rightarrow \mathbb{P}^\pm) \rightarrow \mathbb{P}^\pm,\ \dot{\sqsubseteq}_\pm\rangle$$

Soundness of the abstract sign semantics

- The abstract sign semantics is an abstraction of the collecting property

$$\begin{array}{rrcl} & \mathcal{S}^{\mathbb{C}}[\![\mathsf{A}]\!] & \subseteq & \ddot{\gamma}_{\pm}(\mathcal{S}^{\pm}[\![\mathsf{A}]\!]) \\ \Leftrightarrow & \ddot{\alpha}_{\pm}(\mathcal{S}^{\mathbb{C}}[\![\mathsf{A}]\!]) & \ddot{\sqsubseteq} & \mathcal{S}^{\pm}[\![\mathsf{A}]\!] \end{array}$$

- Precision loss: if the sign of x is $\leqslant 0$ then the sign of x - x is $\top_{\pm}$ not $=0$
- The absolute value is abstracted away
- No precision loss for multiplication $\times$

en.wikipedia.org/wiki/Soundness

Next objective ...

Now that we have defined the collecting semantics $\mathcal{S}^{\mathbb{C}}[\![\mathsf{A}]\!] \in \wp((\mathbb{V} \to \mathbb{Z}) \to \mathbb{Z})$

$$\begin{array}{rcl}
\mathcal{S}^{\mathbb{C}}[\![\mathbf{1}]\!] & = & \{\rho \in (\mathbb{V} \to \mathbb{Z}) \mapsto 1\} \\
\mathcal{S}^{\mathbb{C}}[\![\mathsf{x}]\!] & = & \{\rho \in (\mathbb{V} \to \mathbb{Z}) \mapsto \rho(\mathsf{x})\} \\
\mathcal{S}^{\mathbb{C}}[\![\mathsf{A}_1 - \mathsf{A}_2]\!] & = & \{\rho \in (\mathbb{V} \to \mathbb{Z}) \mapsto f_1(\rho) - f_2(\rho) \mid f_1 \in \mathcal{S}^{\mathbb{C}}[\![\mathsf{A}_1]\!] \wedge f_2 \in \mathcal{S}^{\mathbb{C}}[\![\mathsf{A}_2]\!]\}
\end{array}$$

and the sign abstraction

$$\langle \wp(\mathbb{Z}), \subseteq \rangle \xrightleftharpoons[\alpha_\pm]{\gamma_\pm} \langle \mathbb{P}^\pm, \sqsubseteq \rangle \qquad \text{value properties}$$

$$\langle \wp(\mathbb{V} \to \mathbb{Z}), \subseteq \rangle \xrightleftharpoons[\dot{\alpha}_\pm]{\dot{\gamma}_\pm} \langle \mathbb{V} \to \mathbb{P}^\pm, \dot{\sqsubseteq}_\pm \rangle \qquad \text{environment properties}$$

$$\langle \wp((\mathbb{V} \to \mathbb{Z}) \to \mathbb{Z}), \subseteq \rangle \xrightleftharpoons[\ddot{\alpha}_\pm]{\ddot{\gamma}_\pm} \langle (\mathbb{V} \to \mathbb{P}^\pm) \to \mathbb{P}^\pm, \dot{\sqsubseteq}_\pm \rangle \qquad \text{semantic properties}$$

we are ready to calculate the sign abstract semantics $\mathcal{S}^{\pm}[\![\mathsf{A}]\!] \in (\mathbb{V} \to \mathbb{P}^\pm) \to \mathbb{P}^\pm$ by over approximation of the collecting semantics

$$\ddot{\alpha}_\pm(\mathcal{S}^{\mathbb{C}}[\![\mathsf{A}]\!]) \quad \ddot{\sqsubseteq} \quad \mathcal{S}^{\pm}[\![\mathsf{A}]\!]$$

This sign abstract semantics is a specification of the sign static analyzer.

Calculational design of the sign semantics

Case of a variable x

$$
\begin{array}{ll}
& \ddot{\alpha}_\pm(\mathcal{S}^{\mathbb{C}}[\![\mathsf{x}]\!]) \\
= & \alpha_\pm(\{\mathcal{S}(\rho) \mid \mathcal{S} \in \mathcal{S}^{\mathbb{C}}[\![\mathsf{x}]\!] \wedge \rho \in \dot{\gamma}_\pm(\overset{\pm}{\rho})\}) \hfill \text{(def. (3.34) of } \ddot{\alpha}_\pm\text{)} \\
= & \alpha_\pm(\{\mathcal{A}[\![\mathsf{x}]\!](\rho) \mid \rho \in \dot{\gamma}_\pm(\overset{\pm}{\rho})\}) \hfill \text{(def. (3.13) of } \mathcal{S}^{\mathbb{C}}[\![\mathsf{x}]\!]\text{)} \\
= & \alpha_\pm(\{\rho(\mathsf{x}) \mid \rho \in \dot{\gamma}_\pm(\overset{\pm}{\rho})\}) \hfill \text{(def. (3.4) of } \mathcal{A}[\![\mathsf{x}]\!]\text{)} \\
= & \alpha_\pm(\{\rho(\mathsf{x}) \mid \forall \mathsf{y} \in \mathbb{V} \,.\, \rho(\mathsf{y}) \in \gamma_\pm(\overset{\pm}{\rho}(\mathsf{y}))\}) \hfill \text{(def. (3.24) of } \dot{\gamma}_\pm\text{)} \\
\sqsubseteq & \alpha_\pm(\{\rho(\mathsf{x}) \mid \rho(\mathsf{x}) \in \gamma_\pm(\overset{\pm}{\rho}(\mathsf{x}))\}) \\
& \quad \text{(if y = x, the condition } \rho(\mathsf{x}) \in \gamma_\pm(\overset{\pm}{\rho}(\mathsf{x})) \text{ is the same;} \\
& \quad \text{if y} \neq \text{x the condition } \rho(\mathsf{y}) \in \gamma_\pm(\overset{\pm}{\rho}(\mathsf{y})) \text{ is disgarded;} \\
& \quad \text{So the set } \{\rho(\mathsf{x}) \mid \rho(\mathsf{x}) \in \gamma_\pm(\overset{\pm}{\rho}(\mathsf{x}))\} \text{ is larger and } \alpha_\pm \text{ is increasing)} \\
= & \alpha_\pm(\{x \mid x \in \gamma_\pm(\overset{\pm}{\rho}(\mathsf{x}))\}) \hfill \text{(letting } x = \rho(\mathsf{x})\text{)} \\
= & \alpha_\pm(\gamma_\pm(\overset{\pm}{\rho}(\mathsf{x}))) \hfill \text{(since } S = \{x \mid z \in S\} \text{ for any set } S\text{)} \\
= & \overset{\pm}{\rho}(\mathsf{x}) \hfill \text{(since } \alpha_\pm \circ \gamma_\pm \text{ is the identity)} \\
\triangleq & \mathcal{S}^{\pm}[\![\mathsf{x}]\!]\overset{\pm}{\rho} \hfill \text{(in accordance with (3.21))}
\end{array}
$$

Other cases

- similar for $\ddot{\alpha}_{\pm}(\mathcal{S}^{\mathbb{C}}[\![\mathtt{1}]\!])\overset{\pm}{\rho}$
- by structural induction for $\ddot{\alpha}_{\pm}(\mathcal{S}^{\mathbb{C}}[\![\mathtt{A}_1 \mathtt{-} \mathtt{A}_2]\!])$
- See the book [Cousot, 2021] for more details.

Extension to programs

Automatic static sign program analysis

```
   #include <stdio.h>
      int main () {
      int x;
      scanf("%d",&x);
1:
      while 2: (x>0) {
3:
        x = x-1;
4:
      }
5:    printf("%d\n",x);
      return x;
}
```

What is the sign of x when printing?

Conclusion

Conclusion I

- We have formally defined the semantics of expressions, their properties, their collecting semantics, the sign abstraction, and designed, by calculus, a sign analysis that we have implemented.
- Of course the rule of signs looks trivial, but one can get is wrong! [Sintzoff, 1972]
- The sign analysis is not very precise, but Section **34.11** shows that it is always possible to use infinite abstractions to guarantee more precise results[3].
- For another informal introduction to abstract interpretation, you can read [P. Cousot and R. Cousot, 2010]

[3] *e.g.* Chapter **33** (Static interval analysis) for signs.

Bibliography

Cousot, Patrick (2021). *Principles of Abstract Interpretation*. 1st ed. MIT Press.

P. Cousot and R. Cousot (2010). “A gentle introduction to formal verification of computer systems by abstract interpretation”. In: *Logics and Languages for Reliability and Security*. Ed. by J. Esparza, O. Grumberg, and M. Broy. NATO Science Series III: Computer and Systems Sciences. IOS Press, pp. 1–29.

Sintzoff, Michel (1972). “Calculating Properties of Programs by Valuations on Specific Models”. In: *Proceedings of ACM Conference on Proving Assertions About Programs*. ACM, pp. 203–207.

Home work

Read Ch. **3** "Syntax, semantics, properties, and static analysis of expressions" of

Principles of Abstract Interpretation
Patrick Cousot
MIT Press

The End, Thank you