Compositional Semantics-based Abstract Interpretation

Woosuk Lee

CSE 6049 Program Analysis

Hanyang University, Korea

Goal of This Lecture

- How to instantiate abstract interpretation framework for languages based on a compositional semantics
- Two instances
  - Sign analysis
  - Interval analysis

Language

n	$\in$	$\mathbb{V}$	scalar values
$\mathtt{x}$	$\in$	$\mathbb{X}$	program variables
$\odot$	$::=$	$+ \mid - \mid * \mid \ldots$	binary operators
$\lessdot$	$::=$	$< \mid \leq \mid == \mid \ldots$	comparison operators
E	$::=$		scalar expressions
	$\mid$	n	scalar constant
	$\mid$	$\mathtt{x}$	variable
	$\mid$	$E \odot E$	binary operation
B	$::=$		boolean expressions
	$\mid$	$\mathtt{x} \lessdot n$	comparison of a variable with a constant
C	$::=$		commands
	$\mid$	$\mathbf{skip}$	command that "does nothing"
	$\mid$	$C;\ C$	sequence of commands
	$\mid$	$\mathtt{x} := E$	assignment command
	$\mid$	$\mathbf{input}(\mathtt{x})$	command reading of a value
	$\mid$	$\mathbf{if}(B)\{C\}\mathbf{else}\{C\}$	conditional command
	$\mid$	$\mathbf{while}(B)\{C\}$	loop command

Step 1: Defining Standard Semantics

- Semantic domains

$$\sigma \in \mathbb{M} \stackrel{\text{def}}{=} \mathbb{X} \longrightarrow \mathbb{V} \quad \text{(Memory)}$$

$$n \in \mathbb{V} \stackrel{\text{def}}{=} \mathbb{Z} \qquad \text{(Values = Integers)}$$

- Denotational semantics:

$$[\![E]\!] : \mathbb{M} \longrightarrow \mathbb{V}$$

$$\begin{array}{rcl} [\![n]\!](\sigma) & = & n \\ [\![\mathrm{x}]\!](\sigma) & = & \sigma(\mathrm{x}) \\ [\![E_0 \odot E_1]\!](\sigma) & = & f_{\odot}([\![E_0]\!](\sigma), [\![E_1]\!](\sigma)) \end{array}$$

Function associated to the operator

$$[\![B]\!] : \mathbb{M} \longrightarrow \mathbb{B}$$

$$[\![\mathrm{x} \otimes n]\!](\sigma) \quad = \quad f_{\otimes}(\sigma(\mathrm{x}), n)$$

$$[\![C]\!] : \mathbb{M} \rightarrow \mathbb{M}$$

...

Step 2: Defining Concrete (Collecting) Semantics

$$[\![C]\!]_{\mathscr{P}} : \mathscr{P}(\mathbb{M}) \longrightarrow \mathscr{P}(\mathbb{M})$$

$$\begin{array}{rcl}
[\![\mathtt{skip}]\!]_{\mathscr{P}}(M) & = & M \\
[\![C_0;\ C_1]\!]_{\mathscr{P}}(M) & = & [\![C_1]\!]_{\mathscr{P}}([\![C_0]\!]_{\mathscr{P}}(M)) \\
[\![\mathtt{x} := E]\!]_{\mathscr{P}}(M) & = & \{\sigma[\mathtt{x} \mapsto [\![E]\!](\sigma)] \mid \sigma \in M\} \\
[\![\mathtt{input}(\mathtt{x})]\!]_{\mathscr{P}}(M) & = & \{\sigma[\mathtt{x} \mapsto n] \mid \sigma \in M, n \in \mathbb{V}\} \\
[\![\mathtt{if}(B)\{C_0\}\mathtt{else}\{C_1\}]\!]_{\mathscr{P}}(M) & = & [\![C_0]\!]_{\mathscr{P}}(\mathscr{F}_B(M)) \cup [\![C_1]\!]_{\mathscr{P}}(\mathscr{F}_{\neg B}(M)) \\
[\![\mathtt{while}(B)\{C\}]\!]_{\mathscr{P}}(M) & = & \mathscr{F}_{\neg B}\left(\bigcup_{i \geq 0} ([\![C]\!]_{\mathscr{P}} \circ \mathscr{F}_B)^i (M)\right)
\end{array}$$

Filtering functions

$$\mathscr{F}_B(M) = \{\sigma \in M \mid [\![B]\!](\sigma) = \mathbf{true}\}$$

$$\mathscr{F}_{\neg B}(M) = \{\sigma \in M \mid [\![B]\!](\sigma) = \mathbf{false}\}$$

Loops

- The set of output states of a loop: the infinite union of a family of sets $M_0, M_1, M_n, \ldots$

- where M_i = the output state after running the loop body exactly *i* times

$$M_i = \mathscr{F}_{\neg B}\left(\left(\llbracket C \rrbracket_{\mathscr{P}} \circ \mathscr{F}_B\right)^i (M)\right)$$

Loops

- As a result, the set of output states of the loop is

$$\bigcup_{i \geq 0} M_i = \bigcup_{i \geq 0} \mathscr{F}_{\neg B} \left((\llbracket C \rrbracket_{\mathscr{P}} \circ \mathscr{F}_B)^i (M) \right)$$

- Because $\mathscr{F}_B$ is continuous,

$$\bigcup_{i \geq 0} M_i = \mathscr{F}_{\neg B} \left(\bigcup_{i \geq 0} (\llbracket C \rrbracket_{\mathscr{P}} \circ \mathscr{F}_B)^i (M) \right)$$

Loops

- Alternate definition: Let $F = [\![C]\!]_{\mathscr{P}} \circ \mathscr{F}_B$
  - $M_0 = M;$
  - $M_1 = M \cup F(M) = M \cup F(M_0);$
  - $M_2 = M \cup F(M) \cup F(F(M)) = M \cup F(M \cup F(M))$

  (because F is continuous)

$$\begin{aligned} M_0 &= M \\ M_{k+1} &= M_k \cup F(M_k) \end{aligned}$$

Loops

- Therefore,

$$[\![\texttt{while}(B)\{C\}]\!]_{\mathscr{P}}(M) = \mathscr{F}_{\neg B}(\mathbf{lfp}_M F)$$

$$\text{where } F \triangleq \lambda X.\, M \cup [\![C]\!]_{\mathscr{P}} \circ \mathscr{F}_B(X)$$

Step 3-1: Defining Abstract Domains

- Our goal:

$$\mathscr{P}(\mathbb{M}) \underset{\alpha}{\overset{\gamma}{\leftrightarrows}} \mathbb{A}$$

Abstract Memories

where

$$\mathbb{A} = \mathbb{X} \to \mathbb{A}_{\mathscr{V}}$$

Abstract Mem: Var → Abstract Value

Step 3-1: Defining Abstract Domains

- Abstraction proceeds in two steps:
  - For each variable, we collect the values that this variable may take across a set of states.
  - We over-approximate each of these sets of values with one abstract element per variable using a *value abstraction*.
- Value abstraction:

$$\mathscr{P}(\mathbb{V}) \underset{\alpha_{\mathscr{V}}}{\overset{\gamma_{\mathscr{V}}}{\leftrightarrows}} \mathbb{A}_{\mathscr{V}}$$

Examples of Value Abstractions

- Signs abstraction

$$\mathscr{P}(\mathbb{V}) \underset{\alpha_{\mathscr{S}}}{\overset{\gamma_{\mathscr{S}}}{\leftrightarrows}} \mathbb{A}_{\mathscr{S}}$$

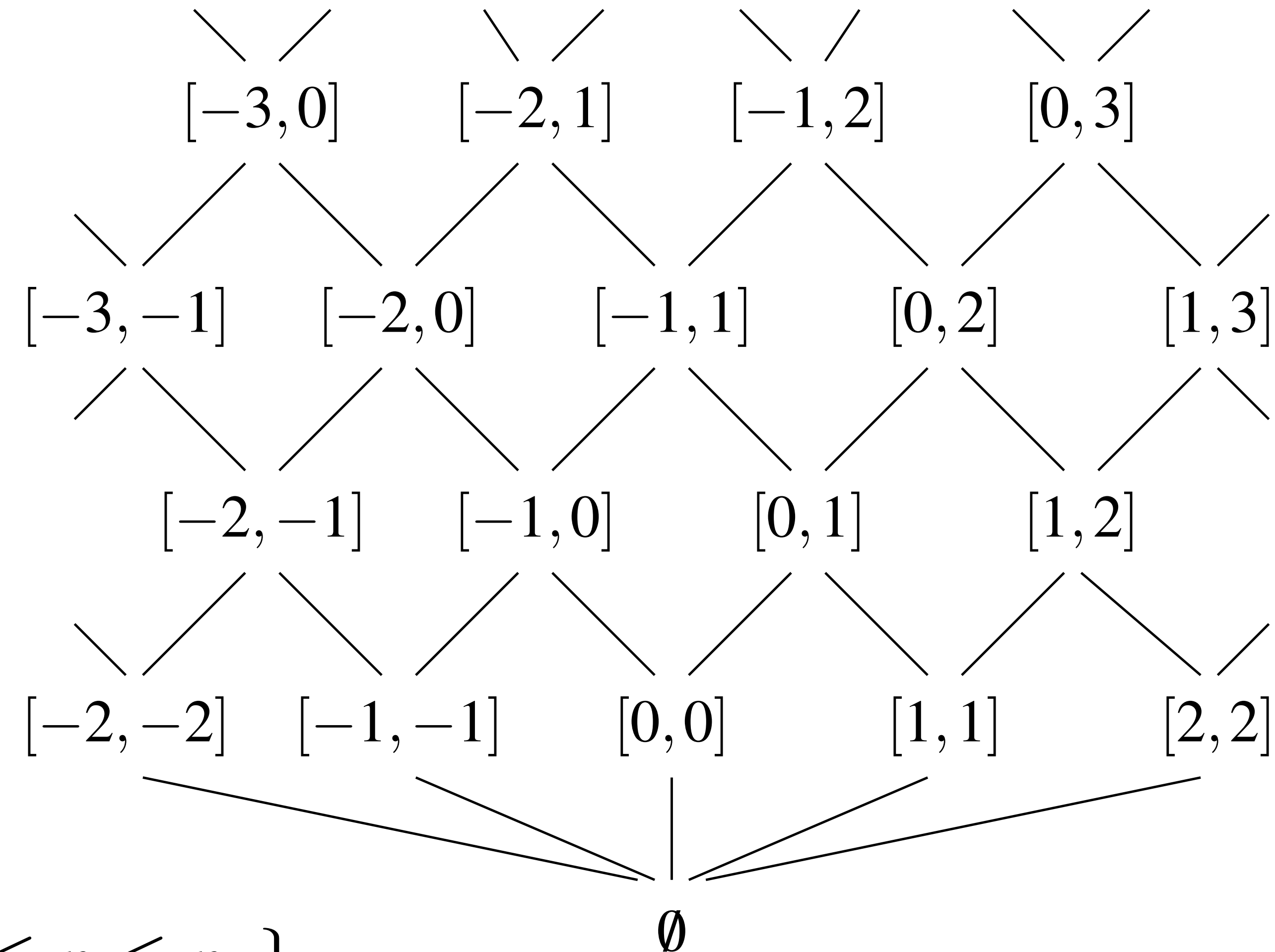
$$\begin{tikzcd}[column sep=small] & \top \arrow[dl, dash] \arrow[dr, dash] & \\ {[\leq 0]} \arrow[dr, dash] & & {[\geq 0]} \arrow[dl, dash] \\ & {[= 0]} \arrow[d, dash] & \\ & \bot & \end{tikzcd}$$

$$\begin{array}{rlcl} \gamma_{\mathscr{S}}: & [\geq 0] & \longmapsto & \{n \in \mathbb{V} \mid n \geq 0\} \\ & [\leq 0] & \longmapsto & \{n \in \mathbb{V} \mid n \leq 0\} \\ & [= 0] & \longmapsto & \{0\} \\ & \top & \longmapsto & \mathbb{V} \\ & \bot & \longmapsto & \emptyset \end{array}$$

Examples of Value Abstractions

- Intervals abstraction

$$\mathscr{P}(\mathbb{V}) \underset{\alpha_{\mathscr{I}}}{\overset{\gamma_{\mathscr{I}}}{\leftrightarrows}} \mathbb{A}_{\mathscr{I}}$$



$$
\begin{array}{rrcl}
\gamma_{\mathscr{I}} : & \perp & \longmapsto & \emptyset \\
& [n_0, n_1] & \longmapsto & \{n \in \mathbb{V} \mid n_0 \leq n \leq n_1\} \\
& [n_0, +\infty) & \longmapsto & \{n \in \mathbb{V} \mid n_0 \leq n\} \\
& (-\infty, n_1] & \longmapsto & \{n \in \mathbb{V} \mid n \leq n_1\} \\
& (-\infty, +\infty) & \longmapsto & \mathbb{V}
\end{array}
$$

Step 3-1: Defining Abstract Domains

- The order relation in $\mathbb{A}$ is defined by the point-wise extension of $\sqsubseteq_{\mathscr{V}}$

$$\forall {M_0}^\sharp, {M_1}^\sharp \in \mathbb{A}.\ {M_0}^\sharp \sqsubseteq {M_1}^\sharp \iff (\forall x \in \mathbb{X}.\ {M_0}^\sharp(x) \sqsubseteq_{\mathscr{V}} {M_1}^\sharp(x))$$

- The least element:

$$\forall x \in \mathbb{X}.\ \perp_{\mathbb{A}}(x) = \perp_{\mathscr{V}}$$

Step 3-1: Defining Abstract Domains

- Then,

$$\begin{array}{rcl}\alpha: M & \longmapsto & (\mathtt{x} \in \mathbb{X}) \longmapsto \alpha_{\mathscr{V}}(\{\sigma(\mathtt{x}) \mid \sigma \in M\}) \\ \gamma: M^\sharp & \longmapsto & \{\sigma \in \mathbb{M} \mid \forall \mathtt{x} \in \mathbb{X},\ \sigma(\mathtt{x}) \in \gamma_{\mathscr{V}}(M^\sharp(\mathtt{x}))\}\end{array}$$

Theorem 1 *If* $\mathscr{P}(\mathbb{V}) \underset{\alpha_{\mathscr{V}}}{\overset{\gamma_{\mathscr{V}}}{\leftrightarrows}} \mathbb{A}_{\mathscr{V}}$ *then* $\mathscr{P}(\mathbb{M}) \underset{\alpha}{\overset{\gamma}{\leftrightarrows}} \mathbb{A}$.

Examples of Memory Abstractions

$\sigma_0:$	$\mathtt{x} \mapsto 25$	$\mathtt{y} \mapsto 7$	$\mathtt{z} \mapsto -12$
$\sigma_1:$	$\mathtt{x} \mapsto 28$	$\mathtt{y} \mapsto -7$	$\mathtt{z} \mapsto -11$
$\sigma_2:$	$\mathtt{x} \mapsto 20$	$\mathtt{y} \mapsto 0$	$\mathtt{z} \mapsto -10$
$\sigma_3:$	$\mathtt{x} \mapsto 35$	$\mathtt{y} \mapsto 8$	$\mathtt{z} \mapsto -9$

- The best abstraction of $\{\sigma_0, \sigma_1, \sigma_2, \sigma_3\}$:
  - With the signs abstraction:

$$M^\sharp : \quad \mathtt{x} \mapsto [\geq 0] \quad \mathtt{y} \mapsto \top \quad \mathtt{z} \mapsto [\leq 0]$$

  - With the intervals abstraction:

$$M^\sharp : \quad \mathtt{x} \mapsto [25, 35] \quad \mathtt{y} \mapsto [-7, 8] \quad \mathtt{z} \mapsto [-12, -9]$$

Step 3-2: Defining Abstract Semantics

If

$$
\begin{tikzcd}
a_{\text{pre}} & \\
\sigma \arrow[u, "\in \gamma(\cdot)"] \arrow[r, "\text{run p}", "(\llbracket \mathtt{p} \rrbracket_{\mathscr{P}})"'] & \sigma'
\end{tikzcd}
$$

then

$$
\begin{tikzcd}
a_{\text{pre}} \arrow[r, "\text{analyze p}", "(\llbracket \mathtt{p} \rrbracket^{\sharp}_{\mathscr{P}})"'] & a_{\text{post}} = \llbracket \mathtt{p} \rrbracket^{\sharp}_{\mathscr{P}}(a_{\text{pre}}) \\
\sigma \arrow[u, "\in \gamma(\cdot)"] \arrow[r, "\text{run p}", "(\llbracket \mathtt{p} \rrbracket_{\mathscr{P}})"'] & \sigma' \arrow[u, "\in \gamma(\cdot)"']
\end{tikzcd}
$$

Our Goal:

Theorem 3.6 (Soundness) *For all command C and all abstract state $M^{\sharp}$, $\llbracket C \rrbracket^{\sharp}_{\mathscr{P}}(M^{\sharp})$ terminates, and:*

$$\llbracket C \rrbracket_{\mathscr{P}}(\gamma(M^{\sharp})) \subseteq \gamma(\llbracket C \rrbracket^{\sharp}_{\mathscr{P}}(M^{\sharp}))$$

Skip

- Bottom element
  - For any command C, $[\![C]\!]_{\mathscr{P}}(\emptyset) = \emptyset$
  - $[\![C]\!]^{\sharp}_{\mathscr{P}}(\bot) = \bot$
- Skip command
  - $[\![\texttt{skip}]\!]^{\sharp}_{\mathscr{P}}(M^{\sharp}) = M^{\sharp}$

Sequence

- Concrete semantics:
  - $[\![\mathrm{p}_0; \mathrm{p}_1]\!]_{\mathscr{P}}(M) = [\![\mathrm{p}_1]\!]_{\mathscr{P}}([\![\mathrm{p}_0]\!]_{\mathscr{P}}(M))$
- Thus, $[\![c_0; c_1]\!]^{\sharp}_{\mathscr{P}}(M^{\sharp}) = [\![c_1]\!]^{\sharp}_{\mathscr{P}}([\![c_0]\!]^{\sharp}_{\mathscr{P}}(M^{\sharp}))$

Abstract Interpretation of Expressions

$$[\![E]\!]^\sharp : \mathbb{A} \longrightarrow \mathbb{A}_{\mathscr{V}}$$

$$\begin{array}{rcl} [\![n]\!]^\sharp(M^\sharp) & = & \phi_{\mathscr{V}}(n) \\ [\![\mathrm{x}]\!]^\sharp(M^\sharp) & = & M^\sharp(\mathrm{x}) \\ [\![E_0 \odot E_1]\!]^\sharp(M^\sharp) & = & f^\sharp_\odot([\![E_0]\!]^\sharp(M^\sharp), [\![E_1]\!]^\sharp(M^\sharp)) \end{array}$$

- $\phi_{\mathscr{V}} : \mathbb{V} \longrightarrow \mathbb{A}_{\mathscr{V}}$: a function that returns an abstraction for a given value (e.g., $\alpha_{\mathscr{S}}(3) = [> 0]$)

- $f^\sharp_\odot : \mathbb{A}_{\mathscr{V}} \times \mathbb{A}_{\mathscr{V}} \longrightarrow \mathbb{A}_{\mathscr{V}}$: approximation of the operator $f_\odot$

Abstract Interpretation of Expressions

- Soundness condition:

$$\text{for all } n_0^\sharp, n_1^\sharp \in \mathbb{A}_{\mathscr{V}},\ \{f_\odot(n_0, n_1) \mid n_0 \in \gamma_{\mathscr{V}}(n_0^\sharp) \text{ and } n_1 \in \gamma_{\mathscr{V}}(n_1^\sharp)\} \subseteq \gamma_{\mathscr{V}}(f_\odot^\sharp(n_0^\sharp, n_1^\sharp))$$

Addition for Signs Abstraction

$f^\sharp_+$	$\bot$	$[\geq 0]$	$[= 0]$	$[\leq 0]$	$\top$
$\bot$	$\bot$	$\bot$	$\bot$	$\bot$	$\bot$
$[\geq 0]$	$\bot$	$[\geq 0]$	$[\geq 0]$	$\top$	$\top$
$[= 0]$	$\bot$	$[\geq 0]$	$[= 0]$	$[\leq 0]$	$\top$
$[\leq 0]$	$\bot$	$\top$	$[\leq 0]$	$[\leq 0]$	$\top$
$\top$	$\top$	$\top$	$\top$	$\top$	$\top$

Subtraction for Signs Abstraction

$f^\sharp_-$	$\bot$	$[\geq 0]$	$[= 0]$	$[\leq 0]$	$\top$
$\bot$	$\bot$	$\bot$	$\bot$	$\bot$	$\bot$
$[\geq 0]$	$\bot$	$\top$	$[\geq 0]$	$[\geq 0]$	$\top$
$[= 0]$	$\bot$	$[\leq 0]$	$[= 0]$	$[\geq 0]$	$\top$
$[\leq 0]$	$\bot$	$[\leq 0]$	$[\leq 0]$	$\top$	$\top$
$\top$	$\top$	$\top$	$\top$	$\top$	$\top$

Abstract Operations for Intervals

- $f^\sharp_+$: $[x_1, x_2] + [y_1, y_2] = [x_1 + y_1, x_2 + y_2]$
- $f^\sharp_-$: $[x_1, x_2] - [y_1, y_2] = [x_1 - y_2, x_2 - y_1]$
- $f^\sharp_\times$:
  $[x_1, x_2] \cdot [y_1, y_2] = [\min\{x_1y_1, x_1y_2, x_2y_1, x_2y_2\}, \max\{x_1y_1, x_1y_2, x_2y_1, x_2y_2\}]$

More cases involving positive/negative infinity …

Example (Interval Operations)

- Suppose we have an abstract memory $M^\sharp$ such that

$M^\sharp(\mathtt{x}) = [10, 20]$ *and* $M^\sharp(\mathtt{y}) = [8, 9]$.

$$
\begin{aligned}
[\![\mathtt{x} + 2 * \mathtt{y} - 6]\!]^\sharp(M^\sharp) &= f^\sharp_-([\![\mathtt{x} + 2 * \mathtt{y}]\!]^\sharp(M^\sharp), [\![6]\!]^\sharp(M^\sharp)) \\
&= f^\sharp_+([\![\mathtt{x}]\!]^\sharp(M^\sharp), [\![2 * \mathtt{y}]\!]^\sharp(M^\sharp)) - [6, 6] \\
&= M^\sharp(\mathtt{x}) + f^\sharp_*([\![2]\!]^\sharp(M^\sharp), [\![\mathtt{y}]\!]^\sharp(M^\sharp)) - [6, 6] \\
&= [10, 20] + [2, 2] * [8, 9] - [6, 6] \\
&= [20, 32]
\end{aligned}
$$

Soundness

Theorem 3.2 (Soundness of the abstract interpretation of expressions) *For all expression* E*, for all non relational abstract element* $M^\sharp$ *and for all memory state* σ *such that* $\sigma \in \gamma(M^\sharp)$*, then:*

$$[\![E]\!](\sigma) \in \gamma([\![E]\!]^\sharp(M^\sharp))$$

Assignments

- Concrete semantics:
  - $[\![\mathtt{x} := E]\!]_{\mathscr{P}}(M) = \{\sigma[\mathtt{x} \mapsto [\![E]\!](\sigma)] \mid \sigma \in M\}$
- Abstract semantics:
  - $[\![\mathtt{x} := E]\!]^{\sharp}_{\mathscr{P}}(M^{\sharp}) = M^{\sharp}[\mathtt{x} \mapsto [\![E]\!]^{\sharp}(M^{\sharp})]$
- Input statement:
  - $[\![\mathbf{\mathtt{input}}(\mathtt{x})]\!]^{\sharp}_{\mathscr{P}}(M^{\sharp}) = M^{\sharp}[\mathtt{x} \mapsto \top_{\mathscr{V}}]$

Example

- Suppose we consider $\mathtt{x} := \mathtt{x} + 2 * \mathtt{y} - 6,$

  $M^\sharp(\mathtt{x}) = [10, 20]$ *and* $M^\sharp(\mathtt{y}) = [8, 9]$.

$$[\![\mathtt{x} := \mathtt{x} + 2 * \mathtt{y} - 6]\!]^\sharp(M^\sharp) = \{\mathtt{x} \mapsto [20, 32], \mathtt{y} \mapsto [8, 9]\}$$

Conditionals

- Concrete semantics:
  - $$[\![\mathtt{if}(B)\{C_0\}\mathtt{else}\{C_1\}]\!]_{\mathscr{P}}(M) = [\![E_0]\!]_{\mathscr{P}}(\mathscr{F}_B(M)) \cup [\![E_1]\!]_{\mathscr{P}}(\mathscr{F}_{\neg B}(M))$$
- Abstract semantics:

Join operator

  - $$[\![\mathtt{if}(B)\{C_0\}\mathtt{else}\{C_1\}]\!]^{\sharp}_{\mathscr{P}}(M^{\sharp}) = [\![C_0]\!]^{\sharp}_{\mathscr{P}}(\mathscr{F}^{\sharp}_B(M^{\sharp})) \sqcup^{\sharp} [\![C_1]\!]^{\sharp}_{\mathscr{P}}(\mathscr{F}^{\sharp}_{\neg B}(M^{\sharp}))$$

Abstract filtering

Abstract Filtering

- Abstract filtering function should satisfy the following soundness condition:

$$\text{for all condition } B, \text{ and for all abstract state } M^\sharp, \ \mathscr{F}_B(\gamma(M^\sharp)) \subseteq \gamma(\mathscr{F}_B^\sharp(M^\sharp))$$

- A trivial example: $\mathscr{F}_B^\sharp(M^\sharp) = \top^\sharp$

Examples of Abstract Filtering

- With the signs abstract domain

$$\mathscr{F}^{\sharp}_{\mathtt{x}<0}(M^{\sharp}) \quad = \quad \begin{cases} (\mathrm{y} \in \mathbb{X}) \longmapsto \bot & \text{if } M^{\sharp}(\mathtt{x}) = [\geq 0] \text{ or } [= 0] \text{ or } \bot \\ M^{\sharp}[\mathtt{x} \mapsto [\leq 0]] & \text{if } M^{\sharp}(\mathtt{x}) = [\leq 0] \text{ or } \top \end{cases}$$

- With the intervals abstract domain if $M^{\sharp}(\mathtt{x}) = [a, b]$

$$\mathscr{F}^{\sharp}_{\mathtt{x}<n}(M^{\sharp}) \quad = \quad \begin{cases} (\mathrm{y} \in \mathbb{X}) \longmapsto \bot & \text{if } a > n \\ M^{\sharp}[\mathtt{x} \mapsto [a, n]] & \text{if } a \leq n \leq b \\ M^{\sharp} & \text{if } b \leq n \end{cases}$$

Analysis of Flow Joins

- The concrete semantics computes the union of the results of both branches.

$$[\![\mathtt{if}(B)\{C_0\}\mathtt{else}\{C_1\}]\!]_{\mathscr{P}}(M) = [\![E_0]\!]_{\mathscr{P}}(\mathscr{F}_B(M)) \boxed{\cup} [\![E_1]\!]_{\mathscr{P}}(\mathscr{F}_{\neg B}(M))$$

- The analysis should over-approximate unions of concrete states.

$$\gamma(M_0^\sharp) \cup \gamma(M_1^\sharp) \subseteq \gamma(M_0^\sharp \sqcup^\sharp M_1^\sharp)$$

- Given the join operator $\sqcup_{\mathscr{V}}^\sharp$ in the value abstract domain, we define the join operator for abstract memories as follows:

$$\text{for all variable } \mathtt{x},\ (M_0^\sharp \sqcup^\sharp M_1^\sharp)(\mathtt{x}) = M_0^\sharp(\mathtt{x}) \sqcup_{\mathscr{V}}^\sharp M_1^\sharp(\mathtt{x})$$

Analysis of Flow Joins

- Example (join operator for intervals)

$$\begin{array}{rcl}[a_0, b_0] \sqcup^{\sharp}_{\mathscr{V}} [a_1, b_1] & = & [\min(a_0, a_1), \max(b_0, b_1)] \\ [a_0, b_0] \sqcup^{\sharp}_{\mathscr{V}} [a_1, +\infty) & = & [\min(a_0, a_1), +\infty)\end{array}$$

- If $\begin{array}{rcl} M_0^{\sharp} & = & [\mathtt{x} \mapsto [0,3]; \mathtt{y} \mapsto [6,7]; \mathtt{z} \mapsto [4,8]] \\ M_1^{\sharp} & = & [\mathtt{x} \mapsto [5,6]; \mathtt{y} \mapsto [0,2]; \mathtt{z} \mapsto [6,9]] \end{array}$

- Then $M_0^{\sharp} \sqcup^{\sharp} M_1^{\sharp} \quad = \quad [\mathtt{x} \mapsto [0,6]; \mathtt{y} \mapsto [0,7]; \mathtt{z} \mapsto [4,9]]$

Theorem 3.4 (Soundness of abstract join) *Let $M_0^{\sharp}$ and $M_1^{\sharp}$ be two abstract states. Then:*

$$\gamma(M_0^{\sharp}) \cup \gamma(M_1^{\sharp}) \subseteq \gamma(M_0^{\sharp} \sqcup^{\sharp} M_1^{\sharp})$$

Example

$M^\sharp$: $\mathtt{x} \mapsto \top, \mathtt{y} \mapsto \top$

$\mathscr{F}^\sharp_{\mathtt{x}>7}(M^\sharp) \quad = \quad M^\sharp[\mathtt{x} \mapsto [8, +\infty)]$

$\mathscr{F}^\sharp_{\mathtt{x}\leq 7}(M^\sharp) \quad = \quad M^\sharp[\mathtt{x} \mapsto (-\infty, 7]]$

$\mathtt{if}(\mathtt{x} > 7)\{$

$\quad \mathtt{y} := \mathtt{x} - 7$

$\}\mathtt{else}\{$

$\quad \mathtt{y} := 7 - \mathtt{x}$

$\}$

$\{\mathtt{x} \mapsto (-\infty, 7], \mathtt{y} \mapsto [0, +\infty)\} \quad \sqcup^\sharp \quad \{\mathtt{x} \mapsto [8, +\infty), \mathtt{y} \mapsto [1, +\infty)\}$

$=$

Final abstract state: $\{\mathtt{x} \mapsto \top, \mathtt{y} \mapsto [0, +\infty)\}$

Loops

- Concrete semantics:

$$[\![\mathtt{while}(B)\{C\}]\!]_{\mathscr{P}}(M) = \mathscr{F}_{\neg B}\left(\bigcup_{i\geq 0}([\![C]\!]_{\mathscr{P}} \circ \mathscr{F}_B)^i(M)\right)$$

- Alternatively,

$$[\![\mathtt{while}(B)\{C\}]\!]_{\mathscr{P}}(M) = \mathscr{F}_{\neg B}(\mathbf{lfp}_M F)$$

$$\text{where } F \triangleq \lambda X.\, M \cup [\![C]\!]_{\mathscr{P}} \circ \mathscr{F}_B(X)$$

- Abstract semantics:

$$[\![\mathtt{while}(B)\{C\}]\!]^\sharp_{\mathscr{P}}(M^\sharp) = \mathscr{F}^\sharp_{\neg B}(\mathbf{lfp}_{M^\sharp} F^\sharp)$$

$$\text{where } F^\sharp \triangleq \lambda X^\sharp.\, M^\sharp \sqcup^\sharp [\![C]\!]^\sharp_{\mathscr{P}} \circ \mathscr{F}^\sharp_B(X^\sharp)$$

Abstract Iterations

$$
\begin{array}{l}
\mathtt{x} := 0; \\
\mathtt{\textbf{while}}(\mathtt{x} \leq 100)\{ \\
\qquad \mathtt{\textbf{if}}(\mathtt{x} \geq 50)\{ \\
\qquad\qquad \mathtt{x} := 10 \\
\qquad \}\mathtt{\textbf{else}}\{ \\
\qquad\qquad \mathtt{x} := \mathtt{x} + 1 \\
\qquad \} \\
\}
\end{array}
$$

$$
\begin{array}{rcl}
M_0^\sharp & = & \{\mathtt{x} \mapsto [0,0]\} \\
M_1^\sharp & = & \{\mathtt{x} \mapsto [0,1]\} \\
M_2^\sharp & = & \{\mathtt{x} \mapsto [0,2]\} \\
\vdots & = & \vdots \\
M_{49}^\sharp & = & \{\mathtt{x} \mapsto [0,49]\} \\
M_{50}^\sharp & = & \{\mathtt{x} \mapsto [0,50]\} \\
M_{51}^\sharp & = & \{\mathtt{x} \mapsto [0,50]\} \\
M_{52}^\sharp & = & \{\mathtt{x} \mapsto [0,50]\} \\
\vdots & = & \vdots
\end{array}
$$

Abstract Semantics

$$
\begin{array}{rcl}
[\![n]\!]^\sharp(M^\sharp) & = & \phi_{\mathscr{V}}(n) \\
[\![\mathtt{x}]\!]^\sharp(M^\sharp) & = & M^\sharp(\mathtt{x}) \\
[\![E_0 \odot E_1]\!]^\sharp(M^\sharp) & = & f^\sharp_\odot([\![E_0]\!]^\sharp(M^\sharp), [\![E_1]\!]^\sharp(M^\sharp)) \\
[\![C]\!]^\sharp_{\mathscr{P}}(\bot) & = & \bot \\
[\![\mathbf{skip}]\!]^\sharp_{\mathscr{P}}(M^\sharp) & = & M^\sharp \\
[\![C_0; C_1]\!]^\sharp_{\mathscr{P}}(M^\sharp) & = & [\![C_1]\!]^\sharp_{\mathscr{P}}([\![C_0]\!]^\sharp_{\mathscr{P}}(M^\sharp)) \\
[\![\mathtt{x} := E]\!]^\sharp_{\mathscr{P}}(M^\sharp) & = & M^\sharp[\mathtt{x} \mapsto [\![E]\!]^\sharp(M^\sharp)] \\
[\![\mathbf{input}(\mathtt{x})]\!]^\sharp_{\mathscr{P}}(M^\sharp) & = & M^\sharp[\mathtt{x} \mapsto \top_{\mathscr{V}}] \\
[\![\mathbf{if}(B)\{C_0\}\mathbf{else}\{C_1\}]\!]^\sharp_{\mathscr{P}}(M^\sharp) & = & [\![C_0]\!]^\sharp_{\mathscr{P}}(\mathscr{F}^\sharp_B(M^\sharp)) \sqcup^\sharp [\![C_1]\!]^\sharp_{\mathscr{P}}(\mathscr{F}^\sharp_{\neg B}(M^\sharp)) \\
[\![\mathbf{while}(B)\{C\}]\!]^\sharp_{\mathscr{P}}(M^\sharp) & = & \mathscr{F}^\sharp_{\neg B}(\mathbf{lfp}_{M^\sharp} F^\sharp) \text{ where } F^\sharp \triangleq \lambda X^\sharp.\, M^\sharp \sqcup^\sharp [\![C]\!]^\sharp_{\mathscr{P}} \circ \mathscr{F}^\sharp_B(X^\sharp)
\end{array}
$$

Soundness Theorems

Theorem 3.1 (Approximation of compositions) *Let* $F_0, F_1 : \mathscr{P}(\mathbb{M}) \longrightarrow \mathscr{P}(\mathbb{M})$ *be two monotone functions, and* $F_0^\sharp, F_1^\sharp : \mathbb{A} \longrightarrow \mathbb{A}$ *be two functions that over-approximate them, that is such that* $F_0 \circ \gamma \subseteq \gamma \circ F_0^\sharp$ *and* $F_1 \circ \gamma \subseteq \gamma \circ F_1^\sharp$*. Then,* $F_0 \circ F_1$ *can be over-approximated by* $F_0^\sharp \circ F_1^\sharp$*.*

Soundness Theorems

Theorem 3.1 (Approximation of compositions) *Let $F_0, F_1 : \mathscr{P}(\mathbb{M}) \longrightarrow \mathscr{P}(\mathbb{M})$ be two monotone functions, and $F_0^\sharp, F_1^\sharp : \mathbb{A} \longrightarrow \mathbb{A}$ be two functions that over-approximate them, that is such that $F_0 \circ \gamma \subseteq \gamma \circ F_0^\sharp$ and $F_1 \circ \gamma \subseteq \gamma \circ F_1^\sharp$. Then, $F_0 \circ F_1$ can be over-approximated by $F_0^\sharp \circ F_1^\sharp$.*

Proof.

if $M^\sharp \in \mathbb{A}$, then $F_1 \circ \gamma(M^\sharp) \subseteq \gamma \circ F_1^\sharp(\mathbb{A})$ (by the soundness assumption on F_1)

$F_0 \circ F_1 \circ \gamma(M^\sharp) \subseteq F_0 \circ \gamma \circ F_1^\sharp(M^\sharp)$ (F_0 is monotone)

$F_0 \circ F_1 \circ \gamma(M^\sharp) \subseteq \gamma \circ F_0^\sharp \circ F_1^\sharp(M^\sharp)$ (by the soundness hypothesis on F_0)

Soundness Theorems

Theorem 3.2 (Soundness of the abstract interpretation of expressions) *For all expression* $\mathtt{E}$*, for all non relational abstract element* $M^\sharp$ *and for all memory state* σ *such that* $\sigma \in \gamma(M^\sharp)$*, then:*

$$[\![E]\!](\sigma) \in \gamma([\![E]\!]^\sharp(M^\sharp))$$

Soundness Theorems

Theorem 3.2 (Soundness of the abstract interpretation of expressions) *For all expression E, for all non relational abstract element $M^\sharp$ and for all memory state σ such that $\sigma \in \gamma(M^\sharp)$, then:*

$$[\![E]\!](\sigma) \in \gamma([\![E]\!]^\sharp(M^\sharp))$$

- Case of constant expressions:
  We assume E is the constant expression defined by the value n. Then, $[\![E]\!](\sigma) = n$, and $[\![E]\!]^\sharp(M^\sharp) = \phi_{\mathscr{V}}(n)$. By definition of the operation $\phi_{\mathscr{V}}$ of the value abstract domain (as stated in Section 3.3.1), $n \in \gamma(\phi_{\mathscr{V}}(n))$, which concludes this case.
- Case of expressions made of a variable:
  We assume E is the expression made of the reading of variable $\mathtt{x}$. Then, $[\![E]\!](\sigma) = \sigma(\mathtt{x})$, and $[\![E]\!]^\sharp(M^\sharp) = M^\sharp(\mathtt{x})$. By assumption, $\sigma \in \gamma(M^\sharp)$, thus, $\sigma(\mathtt{x}) \in \gamma(M^\sharp(\mathtt{x}))$, which concludes this case.
- Case of expressions made of a binary operator applied to two sub-expressions: We assume that E is of the form $E_0 \odot E_1$, where E_0 and E_1 are sub-expressions and $\odot$ is a binary operator. We assume the theorem holds for E_0 and E_1 since we are carrying out the proof by induction over the structure of expressions. Therefore the inductive hypothesis entails that for all $i \in \{0,1\}$, $[\![E_i]\!](\sigma) \in \gamma([\![E_i]\!]^\sharp(M^\sharp))$. Then, $[\![E]\!](\sigma) = f_\odot([\![E_0]\!](\sigma), [\![E_1]\!](\sigma))$ and $[\![E]\!]^\sharp(M^\sharp) = f^\sharp_\odot([\![E_0]\!]^\sharp(M^\sharp), [\![E_1]\!]^\sharp(M^\sharp))$. By the induction hypothesis and by definition of the soundness of the operation of the value abstract domain $f^\sharp_\odot$ (as stated in Section 3.3.1), we have $f_\odot([\![E_0]\!](\sigma), [\![E_1]\!](\sigma)) \in \gamma(f^\sharp_\odot([\![E_0]\!]^\sharp(M^\sharp), [\![E_1]\!]^\sharp(M^\sharp)))$. This concludes the proof of this case.

Soundness Theorems

Theorem 3.3 (Soundness of the abstract interpretation of conditions) *For all expression* B, *for all non relational abstract element* $M^\sharp$ *and for all memory state* σ *such that* $\sigma \in \gamma(M^\sharp)$, *then:*

$$\textit{if } [\![B]\!](\sigma) = \mathbf{true}, \quad \textit{then} \quad \sigma \in \gamma(\mathscr{F}^\sharp_B(M^\sharp))$$

Soundness Theorems

Theorem 3.3 (Soundness of the abstract interpretation of conditions) *For all expression* B*, for all non relational abstract element* $M^\sharp$ *and for all memory state* σ *such that* $\sigma \in \gamma(M^\sharp)$*, then:*

$$\textit{if } [\![B]\!](\sigma) = \mathbf{true}, \quad \textit{then} \quad \sigma \in \gamma(\mathscr{F}^\sharp_B(M^\sharp))$$

Proof. Let B be a condition expression. Let $M^\sharp$ be an abstract state and $\sigma \in \gamma(M^\sharp)$, such that $[\![B]\!](\sigma) = \mathbf{true}$. By definition, the operation $\mathscr{F}^\sharp_B$ of the value abstract domain is assumed to be sound, thus, $\mathscr{F}_B(\gamma(M^\sharp)) \subseteq \gamma(\mathscr{F}^\sharp_B(M^\sharp))$, where $\mathscr{F}_B(M) = \{\sigma \in M \mid [\![B]\!](\sigma) = \mathbf{true}\}$. Since $[\![B]\!](\sigma) = \mathbf{true}$, σ belongs to $\mathscr{F}_B(M)$. This concludes the proof.

Soundness Theorems

Theorem 3.4 (Soundness of abstract join) *Let* $M_0^\sharp$ *and* $M_1^\sharp$ *be two abstract states. Then:*

$$\gamma(M_0^\sharp) \cup \gamma(M_1^\sharp) \subseteq \gamma(M_0^\sharp \sqcup^\sharp M_1^\sharp)$$

Soundness Theorems

Theorem 3.4 (Soundness of abstract join) *Let $M_0^\sharp$ and $M_1^\sharp$ be two abstract states. Then:*

$$\gamma(M_0^\sharp) \cup \gamma(M_1^\sharp) \subseteq \gamma(M_0^\sharp \sqcup^\sharp M_1^\sharp)$$

Proof. We take advantage of the symmetry of both $\cup$ and $\sqcup^\sharp$ so that we simply prove that $\gamma(M_0^\sharp) \subseteq \gamma(M_0^\sharp \sqcup^\sharp M_1^\sharp)$. Let $\sigma \in \gamma(M_0^\sharp)$. To prove that $\sigma \in \gamma(M_0^\sharp \sqcup^\sharp M_1^\sharp)$, we need to establish that, for all variable $\mathtt{x}$, we have $\sigma(\mathtt{x}) \in \gamma_{\mathscr{V}}((M_0^\sharp \sqcup^\sharp M_1^\sharp)(\mathtt{x}))$. By definition of $\sqcup^\sharp$, $(M_0^\sharp \sqcup^\sharp M_1^\sharp)(\mathtt{x}) = M_0^\sharp(\mathtt{x}) \sqcup^\sharp_{\mathscr{V}} M_1^\sharp(\mathtt{x})$. The soundness of $\sqcup^\sharp_{\mathscr{V}}$ guarantees that $\sigma(\mathtt{x}) \in \gamma(M_0^\sharp(\mathtt{x}) \sqcup^\sharp_{\mathscr{V}} M_1^\sharp(\mathtt{x}))$, which concludes the proof.

Soundness Theorems

Theorem 2 (Soundness) *For all command* C *and all abstract state* $M^\sharp$,

$$[\![C]\!]_{\mathscr{P}}(\gamma(M^\sharp)) \subseteq \gamma([\![C]\!]^\sharp_{\mathscr{P}}(M^\sharp))$$

- Case where C is a **skip** statement.
  Then $[\![C]\!]_{\mathscr{P}}(\gamma(M^\sharp)) = \gamma(M^\sharp) = \gamma([\![C]\!]^\sharp_{\mathscr{P}}(M^\sharp))$, so the property trivially holds.
- Case where C is a sequence. We assume the property holds for C_0 and C_1 and prove it for C. Under this assumption Theorem 3.1 applies and proves the property.
- Case where C is an assignment $\mathtt{x} := E$:
  Let $\sigma \in \gamma(M^\sharp)$. We need to prove that $\sigma[\mathtt{x} \mapsto [\![E]\!](\sigma)] \in [\![\mathtt{x} := E]\!]^\sharp_{\mathscr{P}}(M^\sharp) = M^\sharp[\mathtt{x} \mapsto [\![E]\!]^\sharp(M^\sharp)]$. By soundness of the analysis of expressions (Theorem 3.2), we obtain that $[\![E]\!](\sigma) \in \gamma_{\mathscr{V}}([\![E]\!]^\sharp(M^\sharp))$. By definition of γ, that implies the result of the analysis of the assignment is sound.
- Case where C is an input statement $\mathbf{input}(\mathtt{x})$:
  This case is similar to that of a standard assignment; indeed, the only difference is that, in the concrete, $\mathtt{x}$ may get assigned any value, whereas in the abstract, it gets mapped to $\top_{\mathscr{V}}$. We observe that $\top_{\mathscr{V}}$ describes any possible value, so that the argument provided for regular assignment commands applies here in the same way.
- Case where C is the condition statement $\mathbf{if}(B)\{C_0\}\mathbf{else}\{C_1\}$:
  We assume the property holds for C_0 and C_1 and prove it for C:

$$
\begin{array}{rcl}
[\![C]\!]_{\mathscr{P}}(\gamma(M^\sharp)) & = & [\![C_0]\!]_{\mathscr{P}}(\mathscr{F}_B(\gamma(M^\sharp))) \cup [\![C_1]\!]_{\mathscr{P}}(\mathscr{F}_{\neg B}(\gamma(M^\sharp))) \\
& \subseteq & [\![C_0]\!]_{\mathscr{P}}(\gamma(\mathscr{F}^\sharp_B(M^\sharp))) \cup [\![C_1]\!]_{\mathscr{P}}(\gamma(\mathscr{F}^\sharp_{\neg B}(M^\sharp))) \\
& & \quad \text{by soundness of } \mathscr{F}^\sharp_{\cdot} \text{ and monotonicity of } [\![.]\!]_{\mathscr{P}} \\
& \subseteq & \gamma([\![C_0]\!]^\sharp_{\mathscr{P}}(\mathscr{F}^\sharp_B(M^\sharp))) \cup \gamma([\![C_1]\!]^\sharp_{\mathscr{P}}(\mathscr{F}^\sharp_{\neg B}(M^\sharp))) \\
& & \quad \text{by soundness of } [\![C_0]\!]^\sharp_{\mathscr{P}} \text{ and } [\![C_1]\!]_{\mathscr{P}} \text{ (induction hypothesis)} \\
& \subseteq & \gamma([\![C_0]\!]^\sharp_{\mathscr{P}}(\mathscr{F}^\sharp_B(M^\sharp))) \sqcup^\sharp \gamma([\![C_1]\!]^\sharp_{\mathscr{P}}(\mathscr{F}^\sharp_{\neg B}(M^\sharp))) \\
& & \quad \text{by soundness of } \sqcup^\sharp \\
& = & \gamma([\![C]\!]^\sharp(M^\sharp))
\end{array}
$$

Case where C is the while loop $\texttt{while}(B)\{C\}$:

$$[\![\texttt{while}(B)\{C\}]\!]_{\mathscr{P}}(\gamma(M^\sharp)) = \mathscr{F}_{\neg B}(\mathbf{lfp}_{\gamma(M^\sharp)}F)$$

where $F \triangleq \lambda X.\ \gamma(M^\sharp) \cup [\![C]\!]_{\mathscr{P}} \circ \mathscr{F}_B(X)$. And,

$$[\![\texttt{while}(B)\{C\}]\!]^\sharp_{\mathscr{P}}(M^\sharp) = \mathscr{F}^\sharp_{\neg B}(\mathbf{lfp}_{M^\sharp}F^\sharp)$$

where $F^\sharp \triangleq \lambda X^\sharp.\ M^\sharp \sqcup^\sharp [\![C]\!]^\sharp_{\mathscr{P}} \circ \mathscr{F}^\sharp_B(X^\sharp)$.

For any $M^\sharp$, $F(\gamma(M^\sharp)) \subseteq \gamma(F^\sharp(M^\sharp))$ because

$$\begin{array}{rcll} F(\gamma(M^\sharp)) & = & \gamma(M^\sharp) \cup [\![C]\!]_{\mathscr{P}} \circ \mathscr{F}_B(\gamma(M^\sharp)) & \\ \gamma(F^\sharp(M^\sharp)) & = & \gamma(M^\sharp \sqcup^\sharp [\![C]\!]^\sharp_{\mathscr{P}} \circ \mathscr{F}^\sharp_B(M^\sharp)) & \\ & \supseteq & \gamma(M^\sharp) \cup \gamma([\![C]\!]^\sharp_{\mathscr{P}} \circ \mathscr{F}^\sharp_B(M^\sharp)) & \text{(By Theorem 3.4 (Soundness of join))} \end{array}$$

By induction hypothesis $\mathscr{F}^\sharp_B$ and $[\![C]\!]^\sharp_{\mathscr{P}}$ are sound. By Theorem 3.1 (Approximation of compositions),

$$[\![C]\!]_{\mathscr{P}} \circ \mathscr{F}_B \circ \gamma \subseteq \gamma \circ [\![C]\!]^\sharp_{\mathscr{P}} \circ \mathscr{F}^\sharp_B.$$

Therefore, $[\![C]\!]_{\mathscr{P}} \circ \mathscr{F}_B(\gamma(M^\sharp)) \subseteq \gamma([\![C]\!]^\sharp_{\mathscr{P}} \circ \mathscr{F}^\sharp_B(M^\sharp))$ and $F(\gamma(M^\sharp)) \subseteq \gamma(F^\sharp(M^\sharp))$.

From the fixpoint transfer theorem,

$$\mathbf{lfp}F \subseteq \gamma(\mathbf{lfp}F^\sharp)$$

Because $\mathscr{F}^\sharp_{\neg B}$ is sound (i.e., $\mathscr{F}_{\neg B} \circ \gamma \subseteq \gamma \circ \mathscr{F}^\sharp_{\neg B}$), $\mathscr{F}_{\neg B}(\mathbf{lfp}_{\gamma(M^\sharp)}F) \subseteq \gamma \circ \mathscr{F}^\sharp_{\neg B}(\mathbf{lfp}_{M^\sharp}F^\sharp)$ which concludes the proof.

$\square$

What If Loops are Unbounded?

$$
\begin{array}{l}
\mathtt{x} := 0; \\
\mathtt{\textbf{while}}(\mathtt{x} \geq 0)\{ \\
\qquad \mathtt{x} := \mathtt{x} + 1 \\
\}
\end{array}
$$

Signs

$$
\begin{array}{rcl}
M_0^\sharp & = & \{\mathtt{x} \mapsto [= 0]\} \\
M_1^\sharp & = & \{\mathtt{x} \mapsto [\geq 0]\} \\
M_2^\sharp & = & \{\mathtt{x} \mapsto [\geq 0]\}
\end{array}
$$

Intervals

$$
\begin{array}{rcl}
M_0^\sharp & = & \{\mathtt{x} \mapsto [0, 0]\} \\
M_1^\sharp & = & \{\mathtt{x} \mapsto [0, 1]\} \\
M_2^\sharp & = & \{\mathtt{x} \mapsto [0, 2]\} \\
\vdots & = & \vdots \\
M_n^\sharp & = & \{\mathtt{x} \mapsto [0, n]\} \\
\vdots & = & \vdots
\end{array}
$$

What If Bounded Loops Require Too Many Iterations?

```
x : = 0;
while (x < 1000000) {
   x := x + 1
}
```

Needs 1 million iterations to reach a fixpoint for the intervals abstract domain

Widening

- A widening operator over an abstract domain $\mathbb{A}$ is a binary operator ∇, such that

  - For all abstract elements a_0, a_1, we have $\gamma(a_0) \cup \gamma(a_1) \subseteq \gamma(a_0 \nabla a_1)$

  - For all sequence $(a_n)_{n\in\mathbb{N}}$ of abstract elements, the sequence $(a'_n)_{n\in\mathbb{N}}$ defined below is ultimately stationary

$$\begin{cases} a'_0 & = & a_0 \\ a'_{n+1} & = & a'_n \nabla a_n \end{cases}$$

Abstract Iterations with Widening

$\mathtt{abs\_iter}(F^\sharp, M^\sharp)$	$\mathtt{abs\_iter}(F^\sharp, M^\sharp)$
$\quad \mathtt{R} \leftarrow M^\sharp;$	$\quad \mathtt{R} \leftarrow M^\sharp;$
$\quad$ repeat	$\quad$ repeat
$\qquad \mathtt{T} \leftarrow \mathtt{R};$	$\qquad \mathtt{T} \leftarrow \mathtt{R};$
$\qquad \mathtt{R} \leftarrow \mathtt{R} \sqcup^\sharp F^\sharp(\mathtt{R});$	$\qquad \mathtt{R} \leftarrow \mathtt{R} \triangledown F^\sharp(\mathtt{R});$
$\quad$ until $\mathtt{R} = \mathtt{T}$	$\quad$ until $\mathtt{R} = \mathtt{T}$
$\quad$ return $\mathtt{T};$	$\quad$ return $\mathtt{T};$
(a) Iteration with a finite height domain	(b) Iteration with widening and a domain with possibly infinite height

$$[\![\mathbf{\mathtt{while}}(B)\{C\}]\!]^\sharp_{\mathscr{P}}(M^\sharp) \quad = \quad \mathscr{F}^\sharp_{\neg B}(\mathtt{abs\_iter}([\![C]\!]^\sharp_{\mathscr{P}} \circ \mathscr{F}^\sharp_B, M^\sharp))$$

Widening for Intervals

$$
\begin{array}{rclcl}
[a,b] & \triangledown & \perp & = & [a,b] \\
\perp & \triangledown & [c,d] & = & [c,d] \\
[a,b] & \triangledown & [c,d] & = & [(c < a ? -\infty : a), (b < d ? +\infty : b)]
\end{array}
$$

What If Loops are Unbounded? (Revisited)

$$\texttt{x} := 0;$$
$$\texttt{while}(\texttt{x} \geq 0)\{$$
$$\qquad \texttt{x} := \texttt{x} + 1$$
$$\}$$

Intervals

$$\begin{aligned} M_0^\sharp &= \{\texttt{x} \mapsto [0,0]\} \\ M_1^\sharp &= \{\texttt{x} \mapsto [0,+\infty)\} \\ M_2^\sharp &= \{\texttt{x} \mapsto [0,+\infty)\} \end{aligned}$$

How $M_1^\sharp$ was computed?

$\{ \mathsf{x} \longmapsto [0, 0] \} \bigtriangledown \{ \mathsf{x} \longmapsto [1, 1] \}$

$= \{ \mathsf{x} \longmapsto [0, +\infty] \}$

What If Bounded Loops Require Too Many Iterations? (Revisited)

```
x := 0;
while (x < 1000000) {
  x := x + 1
}
```

Intervals

$$\begin{aligned} M_0^\sharp &= \{\mathtt{x} \mapsto [0,0]\} \\ M_1^\sharp &= \{\mathtt{x} \mapsto [0,+\infty)\} \\ M_2^\sharp &= \{\mathtt{x} \mapsto [0,+\infty)\} \end{aligned}$$

- Imprecision occurs: the desirable result is

$$\{\,\mathbf{x} \longmapsto [0,\ 1000000]\,\}$$

- Need to refine the widened result

Narrowing for Intervals

$$
\begin{array}{rcccl}
[a,b] & \triangle & \perp & = & \perp \\
\perp & \triangle & [c,d] & = & \perp \\
[a,b] & \triangle & [c,d] & = & [(a = -\infty ? c : a), (b = +\infty ? d : b)]
\end{array}
$$

What If Bounded Loops Require Too Many Iterations? (Revisited)

```
x : = 0;
while (x < 1000000) {
  x := x + 1
}
```

$\boxed{\textbf{Intervals}}$

$$
\begin{aligned}
M_3{}^\sharp &= \{\mathtt{x} \mapsto [0, +\infty]\} \\
M_4{}^\sharp &= \{\mathtt{x} \mapsto [0, 1000000]\} \\
M_5{}^\sharp &= \{\mathtt{x} \mapsto [0, 1000000]\}
\end{aligned}
$$

Abstract Iterations with Widening & Narrowing

$\texttt{abs\_iter}(F^\sharp, M^\sharp)$
    $\texttt{R} \leftarrow M^\sharp;$
    repeat
        $\texttt{T} \leftarrow \texttt{R};$
        $\texttt{R} \leftarrow \texttt{R} \sqcup^\sharp F^\sharp(\texttt{R});$
    until $\texttt{R} = \texttt{T}$
    return $\texttt{T}$;

(a) Iteration with a finite height domain

$\texttt{abs\_iter}(F^\sharp, M^\sharp)$
    $\texttt{R} \leftarrow M^\sharp;$
    repeat
        $\texttt{T} \leftarrow \texttt{R};$
        $\texttt{R} \leftarrow \texttt{R} \triangledown F^\sharp(\texttt{R});$
    until $\texttt{R} = \texttt{T}$
    repeat
        $\texttt{T} \leftarrow \texttt{R};$
        $\texttt{R} \leftarrow \texttt{R} \vartriangle F^\sharp(\texttt{R});$
    until $\texttt{R} = \texttt{T}$
    return $\texttt{T}$;

(b) Iteration with widening & narrowing and a domain with possibly infinite height

$$[\![\mathtt{while}(B)\{C\}]\!]^\sharp_{\mathscr{P}}(M^\sharp) \quad = \quad \mathscr{F}^\sharp_{\neg B}(\mathtt{abs\_iter}([\![C]\!]^\sharp_{\mathscr{P}} \circ \mathscr{F}^\sharp_B, M^\sharp))$$

Soundness

- The widening and narrowing operators for intervals satisfy the safety conditions for widening and narrowing.

- By the theorems [Widen's safety] and [Narrow's safety], the soundness is guaranteed.