Festschrift on the occasion of Klaus Havelund's 65th birthday

Dynamic interval analysis by abstract interpretation

Patrick Cousot

NYU, New York

pcousot@cs.nyu.edu   cs.nyu.edu/~pcousot

ISoLA 2021, 24 Oct 2021 / Rhodes, Greece

Interval arithmetics

- In scientific computing a real number is represented by a float (floating point number) [IEEE, 1985].
- Because of rounding errors, the floating point computation represents an *uncertain* real computation.
- Ramon E. Moore [Moore, 1966; Moore, Kearfott, and Cloud, 2009] invented "interval arithmetic" to put bounds on rounding errors in floating point computations.
- This guarantees that the *uncertain* real computation is between floating point bounds
- We show that "interval arithmetic" is a sound abstract interpretation of the program semantics (on reals).
- Interval arithmetic is maybe the first dynamic analysis of programs.

`en.wikipedia.org/wiki/Interval_arithmetic`

Prefix trace semantics

Syntax

- We consider a subset of C with variables $\mathtt{x} \in \mathbb{V}$, arithmetic expressions $\mathtt{A} \in \mathbb{A}$, boolean expressions $\mathtt{B} \in \mathbb{B}$, statements $\mathtt{S} \in \mathbb{S}$, statement lists $\mathtt{Sl} \in \mathbb{Sl}$, and programs $\mathtt{P} \in \mathbb{P}$
- By program component $\mathtt{S} \in \mathbb{Pc}$, we mean a statement, statement list , or program
- We axiomatize a labeling of programs to designate program points $\ell \in \mathbb{L}$: $\mathsf{at}[\![\mathtt{S}]\!]$ $\mathsf{after}[\![\mathtt{S}]\!]$ $\mathsf{escape}[\![\mathtt{S}]\!]$ (loop escape via **`break`** ; statement), $\mathsf{break\text{-}to}[\![\mathtt{S}]\!]$, $\mathsf{breaks\text{-}of}[\![\mathtt{S}]\!]$

Trace semantics

- The prefix trace semantics $\mathcal{S}^{*}_{\mathbb{V}}[\![\mathtt{S}]\!]$ of a program component $\mathtt{S}$ is a set of traces describing the beginning of a computation
- The maximal trace semantics are terminated (finite) or nonterminating (infinite) traces $\mathbb{S}^{+\infty}_{\mathbb{V}}$
- A trace π is a finite or infinite sequences of states
- Example: $\langle \ell_1,\ \{\mathtt{x} \rightarrow 1\}\rangle\langle \ell_2,\ \{\mathtt{x} \rightarrow 2\}\rangle\langle \ell_4,\ \{\mathtt{x} \rightarrow 2\}\rangle$
- The states $\langle \ell,\ \rho\rangle \in \mathbb{S}_{\mathbb{V}} \triangleq (\mathbb{L} \times \mathbb{E}\mathrm{v}_{\mathbb{V}})$ are pairs of a label (program point ℓ) and an environment ρ
- Environments $\rho \in \mathbb{E}\mathrm{v}_{\mathbb{V}} \triangleq \mathcal{V} \rightarrow \mathbb{V}$ assign values $\rho(\mathtt{x}) \in \mathbb{V}$ to variables $\mathtt{x} \in \mathcal{V}$
- Values $\mathbb{V}$ can be the set of
  - $\mathbb{R}$ of reals.
  - $\mathbb{F}$ of floats [1]
  - later, $\mathbb{I}$ of float intervals

For simplicity, we assume that execution stops in case of error (e.g. when dividing by zero or returning NaN).

[1] We include ±infinity but exclude NaN, -0, $+0$ for simplicity of the presentation, not hard to handle.

Structural fixpoint definition of the prefix trace semantics

- Iteration statement $\mathsf{S} ::= \texttt{while}\ \ell\ (\mathsf{B})\ \mathsf{S}_b$ (where $\mathsf{at}[\![\mathsf{S}]\!] = \ell$)

$$\boldsymbol{\mathcal{S}}^{*}_{\mathbb{V}}[\![\texttt{while}\ \ell\ (\mathsf{B})\ \mathsf{S}_b]\!] \quad = \quad \mathsf{lfp}^{\subseteq}\ \boldsymbol{\mathcal{F}}^{*}_{\mathbb{V}}[\![\texttt{while}\ \ell\ (\mathsf{B})\ \mathsf{S}_b]\!] \tag{8}$$

$$\begin{aligned}
\boldsymbol{\mathcal{F}}^{*}_{\mathbb{V}}[\![\texttt{while}\ \ell\ (\mathsf{B})\ \mathsf{S}_b]\!]\ X \quad \triangleq \quad & \{\langle \ell,\ \rho\rangle \mid \rho \in \mathbb{E}\mathbb{v}\} && \text{(a)}\\
& \cup \{\pi_2\langle \ell',\ \rho\rangle\langle \mathsf{after}[\![\mathsf{S}]\!],\ \rho\rangle \mid \pi_2\langle \ell',\ \rho\rangle \in X \wedge \boldsymbol{\mathcal{B}}_{\mathbb{V}}[\![\mathsf{B}]\!]\,\rho = \mathsf{ff} \wedge \ell' = \ell\} && \text{(b)}\\
& \cup \{\pi_2\langle \ell',\ \rho\rangle\langle \mathsf{at}[\![\mathsf{S}_b]\!],\ \rho\rangle \cdot \pi_3 \mid \pi_2\langle \ell',\ \rho\rangle \in X \wedge \boldsymbol{\mathcal{B}}_{\mathbb{V}}[\![\mathsf{B}]\!]\,\rho = \mathsf{tt} \wedge && \text{(c)}\\
& \qquad\qquad \langle \mathsf{at}[\![\mathsf{S}_b]\!],\ \rho\rangle \cdot \pi_3 \in \boldsymbol{\mathcal{S}}^{*}_{\mathbb{V}}[\![\mathsf{S}_b]\!] \wedge \ell' = \ell\}
\end{aligned}$$

(a) either the execution observation stop at $[\![\texttt{while}\ \ell\ (\mathsf{B})\ \mathsf{S}_b]\!] = \ell$, or

(b) after a number of iterations, control is back to ℓ, the test is false, and the loop is exited, or

(c) after a number of iterations, control is back to ℓ, the test is true, and the loop body is executed

(This includes the termination of the loop body $\mathsf{after}[\![\mathsf{S}_b]\!] = \mathsf{at}[\![\texttt{while}\ \ell\ (\mathsf{B})\ \mathsf{S}_b]\!] = \ell$)

Maximal trace semantics

- Maximal trace semantics

$$\boldsymbol{\mathcal{S}}^{+}_{\mathbb{V}}[\![\mathsf{S}]\!] \triangleq \{\pi\langle\ell,\ \rho\rangle \in \boldsymbol{\mathcal{S}}^{*}_{\mathbb{V}}[\![\mathsf{S}]\!] \mid (\ell = \mathsf{after}[\![\mathsf{S}]\!]) \vee (\mathsf{escape}[\![\mathsf{S}]\!] \wedge \ell = \mathsf{break\text{-}to}[\![\mathsf{S}]\!])\}$$

$$\boldsymbol{\mathcal{S}}^{\infty}_{\mathbb{V}}[\![\mathsf{S}]\!] \triangleq \lim(\boldsymbol{\mathcal{S}}^{*}_{\mathbb{V}}[\![\mathsf{S}]\!])$$

- Limit

$$\lim \mathscr{T} \triangleq \{\pi \in \mathbb{T}^{\infty} \mid \forall n \in \mathbb{N} \,.\, \pi[0..n] \in \mathscr{T}\}.$$

Float interval abstraction

Float interval domain

- The abstract domain of float intervals is

$$\mathbb{I} \triangleq \begin{array}{l} \{\varnothing\} \cup \{[\underline{x}, \overline{x}] \mid \underline{x}, \overline{x} \in \mathbb{F} \setminus \{-\infty, \infty\} \wedge \underline{x} \leqslant \overline{x}\} \\ \cup \\ \{[-\infty, \overline{x}] \mid \overline{x} \in \mathbb{F} \setminus \{-\infty\}\} \cup \{[\underline{x}, \infty] \mid \underline{x} \in \mathbb{F} \setminus \{\infty\}\} \end{array}$$

  (The intervals $[-\infty, -\infty] \notin \mathbb{I}$ and $[\infty, \infty] \notin \mathbb{I}$ are excluded.)

- The partial order $\sqsubseteq^i$ on $\mathbb{I}$ is interval inclusion $\bot^i \triangleq \varnothing \sqsubseteq^i \bot^i \sqsubseteq^i [\underline{x}, \overline{x}] \sqsubseteq^i [\underline{y}, \overline{y}]$ if and only if $\underline{y} \leqslant \underline{x} \leqslant \overline{x} \leqslant \overline{y}$.

Float notations

- Rounding of real to float:
  - $\overset{\leftarrow}{\rceil\!\rceil} x$ (which can be $-\infty$) is the largest float smaller than or equal to $x \in \mathbb{R}$ (or $\overset{\leftarrow}{\rceil\!\rceil} x = x$ for $x \in \mathbb{F}$)
  - $x \overset{\rightarrow}{\lceil\!\lceil}$ (which can be ∞) is the smallest float greater than or equal to $x \in \mathbb{R}$ (or $x \overset{\rightarrow}{\lceil\!\lceil} = x$ for $x \in \mathbb{F}$).

- Previous and next float:
  - $\overset{\leftarrow}{\rceil} x$ is the largest floating-point number strictly less than $x \in \mathbb{F}$ (which can be $-\infty$)
  - $x \overset{\rightarrow}{\lceil}$ is the smallest floating-point number strictly larger than $x \in \mathbb{F}$ (which can be ∞).

- See the paper for (machine-dependent) soundness conditions for these operations

Float interval abstraction

$$
\begin{array}{rcll}
\alpha^{\mathbb{I}}(x) & \triangleq & [\overset{\leftarrow}{\rrbracket} x, x \overset{\rightarrow}{\llbracket}] & \text{real abstraction by float interval} \quad (14)\\
\gamma^{\mathbb{I}}([\underline{x}, \overline{x}]) & \triangleq & \{x \in \mathbb{R} \mid \underline{x} \leqslant x \leqslant \overline{x}\} & \\
\dot{\alpha}^{\mathbb{I}}(\rho) & \triangleq & \mathrm{x} \in \mathbb{V} \mapsto \alpha^{\mathbb{I}}(\rho(\mathrm{x})) & \text{environment abstraction}\\
\dot{\gamma}^{\mathbb{I}}(\overline{\rho}) & \triangleq & \{\rho \in \mathbb{V} \to \mathbb{R} \mid \forall \mathrm{x} \in \mathbb{V} \,.\, \rho(\mathrm{x}) \in \gamma^{\mathbb{I}}(\overline{\rho}(\mathrm{x}))\} & \\
\ddot{\alpha}^{\mathbb{I}}(\langle \ell,\ \rho\rangle) & \triangleq & \langle \ell,\ \dot{\alpha}^{\mathbb{I}}(\rho)\rangle & \text{state abstraction}\\
\ddot{\gamma}^{\mathbb{I}}(\langle \ell,\ \overline{\rho}\rangle) & \triangleq & \{\langle \ell,\ \rho\rangle \mid \rho \in \dot{\gamma}^{\mathbb{I}}(\overline{\rho})\} & \\
\vec{\alpha}^{\mathbb{I}}(\pi_1 \ldots \pi_n \ldots) & \triangleq & \ddot{\alpha}^{\mathbb{I}}(\pi_1) \ldots \ddot{\alpha}^{\mathbb{I}}(\pi_n) \ldots & \text{[in]finite trace abstraction}\\
\vec{\gamma}^{\mathbb{I}}(\overline{\pi}_1 \ldots \overline{\pi}_n \ldots) & \triangleq & \{\pi_1 \ldots \pi_n \ldots \mid |\pi| = |\overline{\pi}| \wedge \forall i = 1, \ldots, n, \ldots \,.\, \pi_i \in \ddot{\gamma}^{\mathbb{I}}(\overline{\pi_i})\} & \\
\mathring{\alpha}^{\mathbb{I}}(\Pi) & \triangleq & \{\vec{\alpha}^{\mathbb{I}}(\pi) \mid \pi \in \Pi\} & \text{set of traces abstraction}\\
\mathring{\gamma}^{\mathbb{I}}(\overline{\Pi}) & \triangleq & \{\pi \mid \vec{\alpha}^{\mathbb{I}}(\pi) \in \overline{\Pi}\} \quad = \quad \bigcup\{\vec{\gamma}^{\mathbb{I}}(\overline{\pi}) \mid \overline{\pi} \in \overline{\Pi}\} &
\end{array}
$$

Because the floats are a subset of the reals, we can use $\alpha^{\mathbb{I}}$ to abstract both real and float traces (i.e. $\mathbb{V}$ be $\mathbb{R}$ or $\mathbb{F}$).

$$
\langle \wp(\mathbb{S}_{\mathbb{V}}^{+\infty}),\ \subseteq\rangle \underset{\mathring{\alpha}^{\mathbb{I}}}{\overset{\mathring{\gamma}^{\mathbb{I}}}{\leftrightarrows}} \langle \wp(\mathbb{S}_{\mathbb{I}}^{+\infty}),\ \subseteq\rangle \qquad (15)
$$

Float interval arithmetics

Float interval abstraction

- We derive sound abstract operations on float intervals by calculational design (float constants (like 0.1) with rounding, addition $\oplus^i$, subtraction $\ominus^i$, multiplication $\otimes^i$, etc., Boolean comparisons $\olessthan^i$, $\olessthan^i$, etc.
- Subdistributivity $x \otimes^i (y \oplus^i z) \sqsubseteq^i (x \otimes^i y) \oplus^i (x \otimes^i z)$ holds but not distributivity
- Handling tests:
  - real computation: only one branch taken
  - float computation: only one branch taken, but could be the wrong one
  - interval computation: one or both alternatives taken (hence one real trace can be abstracted into interval several traces).
- In most interval arithmetic libraries, this case raises an exception that stops execution, which is a further coarse abstraction of the abstract semantics presented here.

The abstract approximation order

Comparing abstract overapproximations in the concrete

- Program: ℓ_1 $\texttt{x = x - x ;}$ ℓ_2
- Concrete (with precondition x $\in \{-0.1_{\mathbb{R}}, 0.1_{\mathbb{R}}\}$):

$$\Pi = \{\langle \ell_1,\ \texttt{x} = 0.1_{\mathbb{R}}\rangle\langle \ell_2,\ \texttt{x} = 0.0_{\mathbb{R}}\rangle, \quad \langle \ell_1,\ \texttt{x} = -0.1_{\mathbb{R}}\rangle\langle \ell_2,\ \texttt{x} = 0.0_{\mathbb{R}}\rangle\}$$

- Sound abstract semantics on floats:

$$\begin{array}{rcll}
\overline{\Pi}_1 & = & \{\langle \ell_1,\ \texttt{x} = [0.09, 0.11]\rangle\langle \ell_2,\ \texttt{x} = [0.00, 0.00]\rangle, & \Pi \subseteq \mathring{\gamma}^{\mathbb{I}}(\overline{\Pi}_1) \\
 & & \quad \langle \ell_1,\ \texttt{x} = [-0.11, -0.09]\rangle\langle \ell_2,\ \texttt{x} = [0.00, 0.00]\rangle\} & \\
\overline{\Pi}_2 & = & \{\langle \ell_1,\ \texttt{x} = \underbrace{[-0.11, 0.11]}_{\text{input interval}}\rangle\langle \ell_2,\ \texttt{x} = \underbrace{[-0.02, 0.20]}_{\text{interval arithmetic}}\rangle\} & \Pi \subseteq \mathring{\gamma}^{\mathbb{I}}(\overline{\Pi}_2)
\end{array}$$

- Both abstractions are sound, in the concrete, $\Pi \subseteq \mathring{\gamma}^{\mathbb{I}}(\overline{\Pi}_2)$ and $\Pi \subseteq \mathring{\gamma}^{\mathbb{I}}(\overline{\Pi}_2)$
- $\mathring{\gamma}^{\mathbb{I}}(\overline{\Pi}_1)$ is more precise that $\mathring{\gamma}^{\mathbb{I}}(\overline{\Pi}_2)$ since, in the concrete,

$$\mathring{\gamma}^{\mathbb{I}}(\overline{\Pi}_1) \subseteq \mathring{\gamma}^{\mathbb{I}}(\overline{\Pi}_2)$$

- $\overline{\Pi}_1$ and $\overline{\Pi}_2$ are not $\subseteq$-comparable as abstract elements of $\langle \wp(\mathbb{S}_{\mathbb{I}}^{+\infty}),\ \subseteq\rangle$
- So $\subseteq$ does not allow over approximating $\overline{\Pi}_1$ by $\overline{\Pi}_2$!

Sound over-approximation in the concrete

- Define $\overline{\varPi}_1 \overset{\circ}{\sqsubseteq}^{i} \overline{\varPi}_2$

$$\begin{array}{rcl}\overline{\varPi}_1 \overset{\circ}{\sqsubseteq}^{i} \overline{\varPi}_2 & \triangleq & \mathring{\gamma}^{\square}(\overline{\varPi}_1) \subseteq \mathring{\gamma}^{\square}(\overline{\varPi}_2) \\ & = & \forall \overline{\pi}_1 \in \overline{\varPi}_1 \,.\, \forall \pi \in \vec{\gamma}^{\square}(\overline{\pi}_1) \,.\, \exists \overline{\pi}_2 \in \overline{\varPi}_2 \,.\, \pi \in \vec{\gamma}^{\square}(\overline{\pi}_2)\end{array} \tag{16}$$

  to mean that $\overline{\varPi}_1$ is more precise than $\overline{\varPi}_2$, by comparison in the concrete.

- $\overline{\varPi}_1 \subseteq \overline{\varPi}_2$ implies $\overline{\varPi}_1 \overset{\circ}{\sqsubseteq}^{i} \overline{\varPi}_2$ so $\subseteq$ is correct but inadequate for approximation in the abstract (as shown by the previous example)

Sound over-approximation in the abstract

- We express $\mathring{\sqsubseteq}^{i}$ in the abstract, without referring to the concretization $\vec{\gamma}^{\square}$
- We define $\overline{\Pi} \mathring{\sqsubseteq}^{i} \overline{\Pi}'$ so that the traces of $\overline{\Pi}'$ have the same control as the traces of $\overline{\Pi}$ but intervals are larger (and $\overline{\Pi}'$ may contain extra traces due to the imprecision of interval tests).
- $\mathring{\sqsubseteq}^{i}$ is Hoare preorder [Winskel, 1983] on sets of traces.

$$\begin{aligned}
[\underline{x}, \overline{x}] \sqsubseteq^{i} [\underline{y}, \overline{y}] &\triangleq \underline{y} \leqslant \underline{x} \leqslant \overline{x} \leqslant \overline{y} \qquad (18)\\
\rho \mathrel{\dot{\sqsubseteq}^{i}} \rho' &\triangleq \forall \mathrm{x} \in \mathbb{V} \,.\, \rho(\mathrm{x}) \sqsubseteq^{i} \rho'(\mathrm{x})\\
\langle \ell,\ \rho \rangle \mathrel{\ddot{\sqsubseteq}^{i}} \langle \ell',\ \rho' \rangle &\triangleq (\ell = \ell') \wedge (\rho \mathrel{\dot{\sqsubseteq}^{i}} \rho')\\
\overline{\pi} \mathrel{\vec{\sqsubseteq}^{i}} \overline{\pi}' &\triangleq (|\overline{\pi}| = |\overline{\pi}'|) \wedge (\forall i \in [0, |\overline{\pi}|[\,.\, \overline{\pi}_{i} \mathrel{\ddot{\sqsubseteq}^{i}} \overline{\pi}_{i}')\\
\overline{\Pi} \mathrel{\mathring{\sqsubseteq}^{i}} \overline{\Pi}' &\triangleq \forall \overline{\pi} \in \overline{\Pi} \,.\, \exists \overline{\pi}' \in \overline{\Pi}' \,.\, \overline{\pi} \mathrel{\vec{\sqsubseteq}^{i}} \overline{\pi}'
\end{aligned}$$

Lemma 2 $(\overline{\Pi} \mathrel{\mathring{\sqsubseteq}^{i}} \overline{\Pi}') \Rightarrow (\overline{\Pi} \mathrel{\mathring{\sqsubseteq}^{i}} \overline{\Pi}')$. □

Sound over-approximation in the abstract (cont'd)

- Strictly weaker
- Example:

$$\begin{aligned}\overline{\Pi}_1 &= \{\langle \ell_1,\ \mathsf{x} = [0.0, 1.0]\rangle,\\ &\phantom{= \{}\langle \ell_1,\ \mathsf{x} = [1.0, 2.0]\rangle\}\\ \overline{\Pi}_2 &= \{\langle \ell_1,\ \mathsf{x} = [0.0, 0.5]\rangle,\\ &\phantom{= \{}\langle \ell_1,\ \mathsf{x} = [0.5, 2.0]\rangle\}\end{aligned}$$

- $\overline{\Pi}_1 \mathrel{\mathring{\sqsubseteq}^i} \overline{\Pi}_2$ (same concrete traces)
- $\overline{\Pi}_1 \mathrel{\not\mathring{\sqsubseteq}^i} \overline{\Pi}_2$ (no inclusion of abstract traces)
- $\overline{\Pi}_2 \mathrel{\not\mathring{\sqsubseteq}^i} \overline{\Pi}_1$

Soundness and calculational design

- Value (real/float) concrete semantics: $\boldsymbol{\mathcal{S}}^{*}_{\mathbb{V}}\llbracket \mathsf{S} \rrbracket$
- Interval abstract semantics: $\boldsymbol{\mathcal{S}}^{*}_{\mathbb{I}}\llbracket \mathsf{S} \rrbracket$
- Soundness: all value (real/float) traces are included in the interval traces:

$$\begin{array}{rll}
 & \mathring{\alpha}^{\mathbb{I}}(\boldsymbol{\mathcal{S}}^{*}_{\mathbb{V}}\llbracket \mathsf{S} \rrbracket) \mathrel{\mathring{\sqsubseteq}}^{i} \boldsymbol{\mathcal{S}}^{*}_{\mathbb{I}}\llbracket \mathsf{S} \rrbracket & \\
\Rightarrow & \mathring{\alpha}^{\mathbb{I}}(\boldsymbol{\mathcal{S}}^{*}_{\mathbb{V}}\llbracket \mathsf{S} \rrbracket) \mathrel{\mathring{\sqsubseteq}}^{i} \boldsymbol{\mathcal{S}}^{*}_{\mathbb{I}}\llbracket \mathsf{S} \rrbracket & \text{(lemma 2)} \\
\Rightarrow & \mathring{\gamma}^{\mathbb{I}}(\mathring{\alpha}^{\mathbb{I}}(\boldsymbol{\mathcal{S}}^{*}_{\mathbb{V}}\llbracket \mathsf{S} \rrbracket)) \subseteq \mathring{\gamma}^{\mathbb{I}}(\boldsymbol{\mathcal{S}}^{*}_{\mathbb{I}}\llbracket \mathsf{S} \rrbracket) & \text{(def. } \mathring{\sqsubseteq}^{i}\text{)} \\
\Rightarrow & \boldsymbol{\mathcal{S}}^{*}_{\mathbb{V}}\llbracket \mathsf{S} \rrbracket \subseteq \mathring{\gamma}^{\mathbb{I}}(\boldsymbol{\mathcal{S}}^{*}_{\mathbb{I}}\llbracket \mathsf{S} \rrbracket) & \text{(Galois connection } \langle \wp(\mathbb{S}^{+\infty}_{\mathbb{V}}),\ \subseteq\rangle \mathrel{\substack{\xleftarrow{\mathring{\gamma}^{\mathbb{I}}} \\ \xrightarrow[\mathring{\alpha}^{\mathbb{I}}]{}}} \langle \wp(\mathbb{S}^{+\infty}_{\mathbb{I}}),\ \subseteq\rangle,\quad (15)\text{)}
\end{array}$$

- Calculational design:
  - Calculate $\mathring{\alpha}^{\mathbb{I}}(\boldsymbol{\mathcal{S}}^{*}_{\mathbb{V}}\llbracket \mathsf{S} \rrbracket)$
  - Over approximate by $\mathring{\sqsubseteq}^{i}$ to eliminate all concrete operations

Calculational design of the float interval trace semantics

Float interval abstraction of an assignment semantics

- $\mathsf{S} ::= {}^{\ell}\ \mathsf{x} = \mathsf{A}\ ;$
- Concrete semantics on reals ($\mathbb{V} = \mathbb{R}$) or float ($\mathbb{V} = \mathbb{F}$):

$$\begin{aligned}\boldsymbol{\mathcal{S}}^{*}_{\mathbb{V}}[\![\mathsf{S}]\!] \;=\; & \{\langle \ell,\ \rho\rangle \mid \rho \in \mathbb{E}\mathbb{v}_{\mathbb{V}}\} \cup \\ & \{\langle \ell,\ \rho\rangle\langle \mathsf{after}[\![\mathsf{S}]\!],\ \rho[\mathsf{x} \leftarrow \boldsymbol{\mathscr{A}}_{\mathbb{V}}[\![\mathsf{A}]\!]\rho]\rangle \mid \rho \in \mathbb{E}\mathbb{v}_{\mathbb{V}}\}\end{aligned} \tag{2}$$

- Abstract semantics on intervals ($\mathbb{V} = \mathbb{I}$)

$$\begin{aligned}\boldsymbol{\mathcal{S}}^{*}_{\mathbb{I}}[\![\mathsf{S}]\!] \;\triangleq\; & \{\langle \ell,\ \overline{\rho}\rangle \mid \overline{\rho} \in \mathbb{E}\mathbb{v}_{\mathbb{I}}\} \cup \\ & \{\langle \ell,\ \overline{\rho}\rangle\langle \mathsf{after}[\![\mathsf{S}]\!],\ \overline{\rho}[\mathsf{x} \leftarrow \boldsymbol{\mathscr{A}}_{\mathbb{I}}[\![\mathsf{A}]\!]\overline{\rho}]\rangle \mid \overline{\rho} \in \mathbb{E}\mathbb{v}_{\mathbb{I}}\}\end{aligned}$$

- Same traces except for computing on intervals rather than values

Proof

We can now abstract the semantics of real ($\mathbb{V}{=}\mathbb{R}$) or float ($\mathbb{V}{=}\mathbb{F}$) assignments by float intervals.

$$\begin{aligned}
& \alpha^{\mathbb{I}}([^{\ell}\ \mathsf{x} = \mathsf{A}\ ;]) \\
= {} & \{\alpha^{\mathbb{I}}(\pi) \mid \pi \in [^{\ell}\ \mathsf{x} = \mathsf{A}\ ;]\} && \{\!|\ \text{set of traces abstraction (14)}\ |\!\} \\
= {} & \{\alpha^{\mathbb{I}}(\pi) \mid \pi \in \{\langle \ell,\ \rho\rangle \mid \rho \in \mathbb{Ev}_{\mathbb{V}}\} \cup \{\langle \ell,\ \rho\rangle\langle \mathsf{after}\llbracket \mathsf{S}\rrbracket,\ \rho[\mathsf{x} \leftarrow \mathscr{A}_{\mathbb{V}}\llbracket \mathsf{A}\rrbracket\rho]\rangle \mid \rho \in \mathbb{Ev}_{\mathbb{V}}\}\} && \{\!|\ \text{def. } [^{\ell}\ \mathsf{x} = \mathsf{A}\ ;] \text{ in (2)}\ |\!\} \\
= {} & \{\langle \ell,\ \alpha^{\mathbb{I}}(\rho)\rangle \mid \rho \in \mathbb{Ev}_{\mathbb{V}}\} \cup \{\langle \ell,\ \alpha^{\mathbb{I}}(\rho)\rangle\langle \mathsf{after}\llbracket \mathsf{S}\rrbracket,\ \alpha^{\mathbb{I}}(\rho[\mathsf{x} \leftarrow \mathscr{A}_{\mathbb{V}}\llbracket \mathsf{A}\rrbracket\rho])\rangle \mid \rho \in \mathbb{Ev}_{\mathbb{V}}\} && \{\!|\ \text{def. (14) of trace abstraction}\ |\!\} \\
= {} & \{\langle \ell,\ \alpha^{\mathbb{I}}(\rho)\rangle \mid \rho \in \mathbb{Ev}_{\mathbb{V}}\} \cup \{\langle \ell,\ \alpha^{\mathbb{I}}(\rho)\rangle\langle \mathsf{after}\llbracket \mathsf{S}\rrbracket,\ \alpha^{\mathbb{I}}(\rho)[\mathsf{x} \leftarrow \alpha^{\mathbb{I}}(\mathscr{A}_{\mathbb{V}}\llbracket \mathsf{A}\rrbracket\rho])\rangle \mid \rho \in \mathbb{Ev}_{\mathbb{V}}\} && \{\!|\ \text{def. (14) of environment abstraction}\ |\!\} \\
\mathring{\sqsubseteq}^{i} {} & \{\langle \ell,\ \alpha^{\mathbb{I}}(\rho)\rangle \mid \rho \in \mathbb{Ev}_{\mathbb{V}}\} \cup \{\langle \ell,\ \alpha^{\mathbb{I}}(\rho)\rangle\langle \mathsf{after}\llbracket \mathsf{S}\rrbracket,\ \alpha^{\mathbb{I}}(\rho)[\mathsf{x} \leftarrow \mathscr{A}_{\mathbb{I}}\llbracket \mathsf{A}\rrbracket\alpha^{\mathbb{I}}(\rho)]\rangle \mid \rho \in \mathbb{Ev}_{\mathbb{V}}\} && \{\!|\ \text{def. (18) of } \mathring{\sqsubseteq}^{i} \text{ and (21)}\ |\!\} \\
\mathring{\sqsubseteq}^{i} {} & \{\langle \ell,\ \overline{\rho}\rangle \mid \overline{\rho} \in \mathbb{Ev}_{\mathbb{I}}\} \cup \{\langle \ell,\ \overline{\rho}\rangle\langle \mathsf{after}\llbracket \mathsf{S}\rrbracket,\ \overline{\rho}[\mathsf{x} \leftarrow \mathscr{A}_{\mathbb{I}}\llbracket \mathsf{A}\rrbracket\overline{\rho}]\rangle \mid \overline{\rho} \in \mathbb{Ev}_{\mathbb{I}}\} && \{\!|\ \{\alpha^{\mathbb{I}}(\rho) \mid \rho \in \mathbb{Ev}_{\mathbb{V}}\} \subseteq \mathbb{Ev}_{\mathbb{I}} \text{ by (14) for environment abstraction}\ |\!\} \\
\triangleq {} & \boldsymbol{\mathcal{S}}^{*}_{\mathbb{I}}\llbracket^{\ell}\ \mathsf{x} = \mathsf{A}\ ;\rrbracket && \{\!|\ \text{by defining } \boldsymbol{\mathcal{S}}^{*}_{\mathbb{I}}\llbracket^{\ell}\ \mathsf{x} = \mathsf{A}\ ;\rrbracket \text{ as in (2) for } \mathbb{V}{=}\mathbb{I}\ |\!\}
\end{aligned}$$

Approximation $\mathring{\sqsubseteq}^{i}$:

- value $\mathscr{A}_{\mathbb{V}}$ to interval arithmetic $\mathscr{A}_{\mathbb{I}}$
- value to interval environments

Float interval abstraction of an iteration

- Iteration statement $\mathsf{S} ::= \texttt{while}\ \ell\ (\mathsf{B})\ \mathsf{S}_b$ (where $\mathsf{at}[\![\mathsf{S}]\!] = \ell$)

Calculational order

$$\boldsymbol{\mathcal{S}}^{*}_{\mathbb{I}}[\![\texttt{while}\ \ell\ (\mathsf{B})\ \mathsf{S}_b]\!] \quad = \quad \mathsf{lfp}^{\overset{\downarrow}{\subseteq}}\, \boldsymbol{\mathcal{F}}^{*}_{\mathbb{I}}[\![\texttt{while}\ \ell\ (\mathsf{B})\ \mathsf{S}_b]\!] \qquad \text{(8bis)}$$

$$\begin{aligned}
\boldsymbol{\mathcal{F}}^{*}_{\mathbb{I}}[\![\texttt{while}\ \ell\ (\mathsf{B})\ \mathsf{S}_b]\!]\, X \quad \triangleq \quad & \{\langle\ell,\ \rho\rangle \mid \rho \in \mathbb{Ev}_{\mathbb{I}}\} \\
& \cup \{\pi_2\langle\ell',\ \rho\rangle\langle\mathsf{after}[\![\mathsf{S}]\!],\ \rho_{\mathsf{ff}}\rangle \mid \pi_2\langle\ell',\ \rho\rangle \in X \wedge{} \\
& \qquad \exists\overline{\rho}_{\mathsf{tt}}\ .\ \boldsymbol{\mathcal{B}}_{\mathbb{I}}[\![\mathsf{B}]\!]\overline{\rho} = \langle\overline{\rho}_{\mathsf{tt}},\ \overline{\rho}_{\mathsf{ff}}\rangle \wedge \rho_{\mathsf{ff}} \neq \dot{\varnothing} \wedge \ell' = \ell\} \\
& \cup \{\pi_2\langle\ell',\ \rho\rangle\langle\mathsf{at}[\![\mathsf{S}_b]\!],\ \rho_{\mathsf{tt}}\rangle\pi_3 \mid \pi_2\langle\ell',\ \rho\rangle \in X \wedge{} \\
& \qquad \exists\overline{\rho}_{\mathsf{ff}}\ .\ \boldsymbol{\mathcal{B}}_{\mathbb{I}}[\![\mathsf{B}]\!]\overline{\rho} = \langle\overline{\rho}_{\mathsf{tt}},\ \overline{\rho}_{\mathsf{ff}}\rangle \wedge \rho_{\mathsf{tt}} \neq \dot{\varnothing} \wedge{} \\
& \qquad \langle\mathsf{at}[\![\mathsf{S}_b]\!],\ \rho_{\mathsf{tt}}\rangle\pi_3 \in \boldsymbol{\mathcal{S}}^{*}_{\mathbb{I}}[\![\mathsf{S}_b]\!] \wedge \ell' = \ell\}
\end{aligned}$$

Approximation order

- Soundness $\mathring{\alpha}^{\mathbb{I}}(\boldsymbol{\mathcal{S}}^{*}_{\mathbb{V}}[\![\mathsf{S}]\!]) \ \overset{\downarrow}{\dot{\sqsubseteq}^{i}}\ \boldsymbol{\mathcal{S}}^{*}_{\mathbb{I}}[\![\mathsf{S}]\!]$
- Only other example is Mycroft's strictness analysis (computational order $\sqsubseteq$ and approximation order $\subseteq$))

Specification of an implementation

- The abstraction to a transition system provides a small-step operational semantics of the program (specifying an implementation)
- We used trace abstractions so there is no need for [bi-]simulations, etc. in the proof of correctness of the implementation

Summary

Summary

- We have defined the value semantics $\mathcal{S}^*_{\mathbb{V}}$ of the language for reals and floats (executions on reals are not implementable/too costly to implement[2])
- Next, we define the interval abstraction $\mathring{\alpha}^{\mathbb{I}}$ of a value semantics (replacing reals by float intervals)
- The best float interval semantics of the value semantics is $\mathring{\alpha}^{\mathbb{I}}(\mathcal{S}^*_{\mathbb{V}})$ (execute on reals and then abstract to float intervals, not implementable)
- We define a sound over-approximation partial order $\mathring{\sqsubseteq}^{i}$ of interval semantics (with larger intervals)
- Next, we calculate the interval semantics $\mathcal{S}^*_{\mathbb{I}}$ of the language (executions on float intervals)
- By calculational design $\mathring{\alpha}^{\mathbb{I}}(\mathcal{S}^*_{\mathbb{V}}) \mathrel{\mathring{\sqsubseteq}^{i}} \mathcal{S}^*_{\mathbb{I}}$, so the interval semantics is a sound abstraction of the value semantics
- Abstraction to a transition system formalizes the soundness of the implementation

[2] e.g. using Bill Gosper's exact algorithms for continued fraction arithmetic.

Conclusion

Conclusion

- Interval arithmetics in scientific computing put bounds on rounding errors in floating point arithmetic [Moore, 1966].
- It is an abstract interpretation of the trace semantics and can be computed at runtime for one trace at a time.
- Tests may have to consider many executions, which can be quite inefficient (and often considered an error in practice).
- A further abstract yields the static interval analysis (by joining states on paths at each program point to get invariants).
- More generally, this provides a framework for dynamic analysis (their static over approximation, and the combination of the two).
- This general abstract interpretation framework for dynamic analysis is described in the paper (interval arithmetic is an instance)
- Soundness guarantee!

The End, Thank you