New York University, CIMS, CS, Course CSCI-GA.3140-001, Spring 2024

"Abstract Interpretation"

Ch. 6, Structural Deductive Stateless Prefix Trace Semantics

Patrick Cousot

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

Class 2, Monday, January 29th, 2024, 4:55-6:55 PM, WWH, Room CIWW 202

These slides are available at
http://cs.nyu.edu/~pcousot/courses/spring24/CSCI-GA.3140-001/slides/02--2024-01-30-syntax-semantics-traces-oo/slides-06--prefix-trace-semantics-AI.pdf

Chapter 6

Ch. 6, Structural Deductive Stateless Prefix Trace Semantics

Trace semantics, informally

Hand computation of

$$\underbrace{\underbrace{\underbrace{(1-1)}_{a}-1}_{b} < \underbrace{(1-1)}_{a}}_{c}$$

is

$a = 1 - 1$
$a = 0$
$b = a - 1$
$b = 0 - 1$
$b = -1$
$c = b < a$
$c = -1 < 0$
$c = tt$

read from

partial trace

maximal finite trace

Syntax and trace semantics of a language

- syntax: rules to write programs of the language;
- semantics: defines the runtime behavior of programs that is what and how they compute when executed:
  - trace: sequence of events recording the actions executed during a program execution,
  - partial trace: finite observation of an execution; this observation can stop at any time,
  - finite (maximal) trace: partial trace that ends upon execution termination,
  - infinite trace: infinite observation of an execution that never terminates,
  - maximal trace: finite or infinite execution trace.

Traces

Finite traces of a program: P

- Program:

$$\begin{array}{l} \ell_1 \; \mathtt{x = x + 1 \; ;} \\ \quad \mathbf{\mathtt{while}} \; \ell_2 \; (\mathtt{tt}) \; \{ \\ \qquad \ell_3 \; \mathtt{x = x + 1 \; ;} \\ \qquad \mathbf{\mathtt{if}} \; \ell_4 \; (\mathtt{x > 2}) \; \ell_5 \; \mathbf{\mathtt{break}} \; ;\} \ell_6 ; \ell_7 \end{array} \tag{4.5}$$

- Prefix traces (from ℓ_1, initially $\mathtt{x} = 0$):
  - ℓ_1
  - $$\ell_1 \xrightarrow{\mathtt{x = x + 1 = 1}} \ell_2 \xrightarrow{\mathtt{tt}} \ell_3 \xrightarrow{\mathtt{x = x + 1 = 2}} \ell_4 \xrightarrow{\neg(\mathtt{x > 2})} \ell_2 \xrightarrow{\mathtt{tt}} \ell_3 \tag{6.2}$$

- Finite (maximal) traces:
  - $$\ell_1 \xrightarrow{\mathtt{x = x + 1 = 1}} \ell_2 \xrightarrow{\mathtt{tt}} \ell_3 \xrightarrow{\mathtt{x = x + 1 = 2}} \ell_4 \xrightarrow{\neg(\mathtt{x > 2})} \ell_2 \xrightarrow{\mathtt{tt}} \ell_3 \xrightarrow{\mathtt{x = x + 1 = 3}}$$
    $$\ell_4 \xrightarrow{\mathtt{x > 2}} \ell_5 \xrightarrow{\text{break}} \ell_6 \xrightarrow{\text{skip}} \ell_7$$

Infinite traces of a program: P

- Program:

$$\ell_1\ \mathtt{x = 0}\ ;\ \mathbf{while}\ \ell_2\ (\mathrm{tt})\ \{\ \ell_3\ \mathtt{x = x + 1}\ ;\ \}\ \ell_4$$

- Infinite trace:

$$\ell_1 \xrightarrow{\mathtt{x = 0} = 0} \ell_2 \xrightarrow{\mathrm{tt}} \ell_3 \xrightarrow{\mathtt{x = x + 1} = 1} \ell_2 \xrightarrow{\mathrm{tt}} \ell_3 \xrightarrow{\mathtt{x = x + 1} = 2} \ell_2 \ldots \ell_2 \xrightarrow{\mathrm{tt}} \ell_3$$

$$\xrightarrow{\mathtt{x = x + 1} = n} \ell_2 \xrightarrow{\mathrm{tt}} \ell_3 \xrightarrow{\mathtt{x = x + 1} = n + 1} \ell_2 \ldots$$

Traces

- $\mathbb{T}^{+}$: the set of all finite traces,
- $\mathbb{T}^{\infty}$: the set of all infinite traces,
- $\mathbb{T}^{+\infty}$: the set of all finite or infinite traces.
- Conventions:
  - we write $\pi = {}^{\ell}\pi'$ to make clear that the trace π is assumed to start with the program label ℓ (although π' is not itself a properly formed trace),
  - we write $\pi = \pi'^{\ell}$ when assuming that the trace π is finite and ends with label ℓ (although, again, π' is not itself a properly formed trace).

Trace concatenation $\frown$

- Definition:

$$
\begin{array}{rcll}
\pi_1{\ell_1} \frown {\ell_2}\pi_2 & & & \text{undefined if } \ell_1 \neq \ell_2 \\
\pi_1{\ell_1} \frown {\ell_1}\pi_2 & \triangleq & \pi_1{\ell_1}\pi_2 & \text{if } \pi_1 \text{ is finite} \\
\pi_1 \frown \pi_2 & \triangleq & \pi_1 & \text{if } \pi_1 \text{ is infinite}
\end{array}
$$

- In pattern matching, we sometimes need the empty trace $\ni$. For example $\ell\pi\ell' = \ell$ then $\pi = \ni$ and $\ell = \ell'$.

Value of variables

Values of variables on a trace

- the value $\varrho(\pi)\mathsf{x}$ of variable x *at the end* of trace π is the last value assigned to x (or 0 at initialization).

$$
\begin{aligned}
\varrho(\pi^{\ell} \xrightarrow{\mathsf{x}\,=\,\mathsf{A}\,=\,\upsilon} \ell')\mathsf{x} &\triangleq \upsilon \\
\varrho(\pi^{\ell} \xrightarrow{\cdots} \ell')\mathsf{x} &\triangleq \varrho(\pi^{\ell})\mathsf{x} \quad \text{otherwise} \\
\varrho(\ell)\mathsf{x} &\triangleq 0
\end{aligned}
\tag{6.6}
$$

Prefix trace semantics of a statement

Prefix trace semantics

- Let $\pi_1\text{at}[\![S]\!]$ be an initialization trace ending on entry $\text{at}[\![S]\!]$ of statement S.
- $\boldsymbol{\mathcal{S}}^*[\![S]\!](\pi_1\text{at}[\![S]\!])$ is the set of prefix traces $\text{at}[\![S]\!]\pi_2\ell$ of S continuing the trace $\pi_1\text{at}[\![S]\!]$ and reaching some program label $\ell \in \text{labx}[\![S]\!]$.
- Schematically,

$$\xrightarrow{\quad\pi_1\quad} \underbrace{\text{at}[\![S]\!] \xrightarrow{\quad\pi_2\quad} \ell}_{\in\ \boldsymbol{\mathcal{S}}^*[\![S]\!](\pi_1\text{at}[\![S]\!])}$$

- Although our language is determinist, we consider a set of possible continuations to cope e.g. with inputs and random number generation.
- By convention $\boldsymbol{\mathcal{S}}^*[\![S]\!](\pi_1\ell) = \varnothing$ when $\ell \neq \text{at}[\![S]\!]$.

Maximal finite trace semantics

- Let $\pi_1\text{at}[\![\mathsf{S}]\!]$ be an initialization trace ending on entry $\text{at}[\![\mathsf{S}]\!]$ of statement S.
- $\boldsymbol{\mathcal{S}}^{+}[\![\mathsf{S}]\!](\pi_1\text{at}[\![\mathsf{S}]\!])$ is the set of maximal finite traces $\text{at}[\![\mathsf{S}]\!]\pi_2\text{after}[\![\mathsf{S}]\!]$ of S continuing the trace $\pi_1\text{at}[\![\mathsf{S}]\!]$ and reaching $\text{after}[\![\mathsf{S}]\!]$.
- Schematically,

$$\xrightarrow{\quad\pi_1\quad}\underbrace{\text{at}[\![\mathsf{S}]\!]\xrightarrow{\quad\pi_2\quad}\text{after}[\![\mathsf{S}]\!]}_{\in\ \boldsymbol{\mathcal{S}}^{+}[\![\mathsf{S}]\!](\pi_1\text{at}[\![\mathsf{S}]\!])}$$

- Formally,

$$\boldsymbol{\mathcal{S}}^{+}[\![\mathsf{S}]\!](\pi_1\text{at}[\![\mathsf{S}]\!]) \quad\triangleq\quad \{\pi_2\ell \in \boldsymbol{\mathcal{S}}^{*}[\![\mathsf{S}]\!](\pi_1\text{at}[\![\mathsf{S}]\!]) \mid \ell = \text{after}[\![\mathsf{S}]\!]\} \tag{6.9}$$

Introduction to rule-based structural definitions

Structural definitions and proofs

- Structural definitions are recursive definitions over the syntax of programs;
- Structural proofs generalize proofs by recurrence to induction on the syntax of programs;
- Structural proofs are well suited to prove properties of structural definitions (e.g. that a structural definition is well-defined i.e. the recursive definition considered as a program does terminate).

Example of rule-based structural definition

- Denotation of positive integers $\mathbb{N}^+$ by a collection of sticks:

$$\mathbb{N}^+ ::= \mathbf{I} \mid \mathbb{N}^+\mathbf{I}$$

- Example: **IIIIII** is six
- Structural definition of the set $\mathbb{O}$ of odd positive integers:
  - axiom $\dfrac{}{\mathbf{I} \in \mathbb{O}}$
  - inference rule $\dfrac{n \in \mathbb{O}}{n\mathbf{II} \in \mathbb{O}}$
- Set $s(n)$ of numbers smaller than or equal to n:

$$\frac{}{n \in s(n)} \qquad \frac{m\mathbf{I} \in s(n)}{m \in s(n)}$$

Example: $\mathbf{III} \in s(\mathbf{III})$ by the axiom so $\mathbf{II} \in s(\mathbf{III})$ by the inference rule so $\mathbf{I} \in s(\mathbf{III})$ by the inference rule proving that $s(\mathbf{III}) = \{\mathbf{III}, \mathbf{II}, \mathbf{I}\}$.

Structural prefix trace semantics

Prefix trace semantics $\widehat{\mathcal{S}}^*[\![\mathsf{S}]\!]$ of a program component S

$$\pi_2 \in \widehat{\mathcal{S}}^*[\![\mathsf{S}]\!](\pi_1)$$

- the prologue trace π_1 terminates at $\mathsf{at}[\![\mathsf{S}]\!]$
- the continuation trace π_2 starts at $\mathsf{at}[\![\mathsf{S}]\!]$

(will be proved by structural induction on S)

Axioms

Structural prefix trace semantics at a program component

Prefix trace at a program component S

- $$\frac{}{\mathsf{at}[\![\mathsf{S}]\!] \in \widehat{\boldsymbol{\mathcal{S}}}^*[\![\mathsf{S}]\!](\pi_1\mathsf{at}[\![\mathsf{S}]\!])} \tag{6.11}$$

A prefix continuation of the traces $\pi_1\mathsf{at}[\![\mathsf{S}]\!]$ arriving at a program, statement or statement list S can be reduced to the program point $\mathsf{at}[\![\mathsf{S}]\!]$ at this program, statement or statement list S.

Structural prefix trace semantics of an empty statement list

Prefix traces of an empty statement list $\texttt{Sl} ::= \epsilon$

- $$\frac{}{\text{at}[\![\texttt{Sl}]\!] \in \widehat{\mathcal{S}}^*[\![\texttt{Sl}]\!](\pi\text{at}[\![\texttt{Sl}]\!])} \tag{6.15}$$

- A prefix/maximal trace π of the empty statement list ϵ continuing some trace is reduced to the program label $\text{at}[\![\texttt{Sl}]\!]$ at that empty statement.
- This case is redundant and already covered by (6.11).

Structural prefix trace semantics of an assignment statement

Prefix traces of an assignment statement $\mathtt{S} ::= \ell\ \mathtt{x} = \mathtt{A}\ ;$

- $$\frac{v = \mathscr{A}[\![\mathtt{A}]\!]\varrho(\pi^\ell)}{\ell \xrightarrow{\mathtt{x} = \mathtt{A} = v} \mathsf{after}[\![\mathtt{S}]\!] \in \widehat{\boldsymbol{\mathcal{S}}}^*[\![\mathtt{S}]\!](\pi^\ell)} \tag{6.16}$$

A prefix/maximal finite trace of an assignment $\ell\ \mathtt{x} = \mathtt{E}\ ;$ continuing some trace π^ℓ is ℓ followed by the event $\mathtt{x} = v$ where v is the last value of $\mathtt{x}$ previously assigned to $\mathtt{x}$ on π^ℓ (otherwise initialized to 0) and finishing at the label $\mathsf{after}[\![\mathtt{S}]\!]$ after the assignment.

Structural prefix trace semantics of a skip statement

Prefix traces of a skip statement $\mathtt{S} ::= \ell\,;$

- $$\frac{}{\ell \xrightarrow{\ \mathtt{skip}\ } \mathsf{after}[\![\mathtt{S}]\!] \in \widehat{\boldsymbol{\mathcal{S}}}^{*}[\![\mathtt{S}]\!](\pi^{\ell})} \tag{6.17}$$

A prefix/maximal finite trace of a skip statement ℓ; continuing an initial trace π^{ℓ} arriving at ℓ is just continuing after the skip statement.

Structural prefix trace semantics of a break statement

Prefix traces of a break statement $\mathsf{S} ::= {}^{\ell}\ \mathtt{break}\ ;$

- $$\frac{}{\ell \xrightarrow{\text{break}} \text{break-to}[\![\mathsf{S}]\!] \in \widehat{\boldsymbol{\mathcal{S}}}^{*}[\![\mathsf{S}]\!](\pi^{\ell})} \tag{6.29}$$

A prefix/maximal finite trace of a break ${}^{\ell}$ $\mathtt{break}$; continuing some initial trace π^{ℓ} is the trace ℓ followed by the $\mathtt{break}$; event and ending at the break label break-to$[\![\mathsf{S}]\!]$ (which is the exit label of the closest enclosing iteration loop or else the program exit).

Structural inference rules

Structural prefix trace semantics of a program

Prefix traces of a program $\mathtt{P} ::= \mathtt{Sl}\ \ell$

- $$\frac{\pi_2 \in \widehat{\boldsymbol{\mathcal{S}}}^*[\![\mathtt{Sl}]\!](\pi_1\mathsf{at}[\![\mathtt{Sl}]\!])}{\pi_2 \in \widehat{\boldsymbol{\mathcal{S}}}^*[\![\mathtt{P}]\!](\pi_1\mathsf{at}[\![\mathtt{P}]\!])} \tag{6.12}$$

If $\mathtt{P} ::= \mathtt{Sl}\ \ell$ then the prefix continuations of the traces $\pi_1\mathsf{at}[\![\mathtt{Sl}]\!]$ arriving at program entry $\mathsf{at}[\![\mathtt{P}]\!] = \mathsf{at}[\![\mathtt{Sl}]\!]$ are the continuations of the statement list $\mathtt{Sl}$.

Structural prefix trace semantics of a compound statement

Prefix traces of a compound statement `S ::= { Sl }`

- $$\frac{\pi_2 \in \widehat{\boldsymbol{\mathcal{S}}}^*[\![\mathtt{Sl}]\!](\pi_1)}{\pi_2 \in \widehat{\boldsymbol{\mathcal{S}}}^*[\![\mathtt{S}]\!](\pi_1)} \tag{6.30}$$

A prefix trace of a compound statement `{ Sl }` is that of its statement list `Sl`.

Structural prefix trace semantics of a conditional statement

Prefix traces of a conditional statement $\mathtt{S} ::= \mathtt{if}\ \ell\ (\mathtt{B})\ \mathtt{S}_t$

- $$\frac{\mathscr{B}[\![\mathtt{B}]\!]\varrho(\pi_1\ell) = \mathrm{ff}}{\ell \xrightarrow{\neg(\mathtt{B})} \mathrm{after}[\![\mathtt{S}]\!] \in \widehat{\mathcal{S}}^*[\![\mathtt{S}]\!](\pi_1\ell)} \qquad (6.18)$$

- $$\frac{\mathscr{B}[\![\mathtt{B}]\!]\varrho(\pi_1\ell) = \mathrm{tt}, \quad \pi_2 \in \widehat{\mathcal{S}}^*[\![\mathtt{S}_t]\!](\pi_1\ell \xrightarrow{\mathtt{B}} \mathrm{at}[\![\mathtt{S}_t]\!])}{\ell \xrightarrow{\mathtt{B}} \mathrm{at}[\![\mathtt{S}_t]\!] \frown \pi_2 \in \widehat{\mathcal{S}}^*[\![\mathtt{S}]\!](\pi_1\ell)} \qquad (6.19)$$

Structural prefix trace semantics of a conditional statement

- A prefix trace of a conditional statement $\texttt{if}\ \ell\ (\mathsf{B})\ \mathsf{S}_t$ continuing some initial trace $\pi_1{}^\ell$ is
  - either ℓ (a case already covered by (6.11));
  - or, in case (6.18), ℓ followed by the event $\neg(\mathsf{B})$ when the value of this boolean expression on $\pi_1{}^\ell$ is ff and finishing at the label after$[\![\mathsf{S}]\!]$ after the conditional statement;
  - or, in case case (6.19). when the value of the boolean expression B on $\pi_1{}^\ell$ is tt, ℓ followed by the test event B followed by a prefix trace of S_t continuing $\pi_1{}^\ell \xrightarrow{\mathsf{B}}$ at$[\![\mathsf{S}_t]\!]$.

Structural prefix trace semantics of a conditional statement

Prefix traces of a conditional statement $\mathtt{S} ::= \mathtt{if}\ \ell\ (\mathtt{B})\ \mathtt{S}_t\ \mathtt{else}\ \mathtt{S}_f$

- $$\frac{\mathscr{B}[\![\mathsf{B}]\!]\varrho(\pi_1\ell) = \mathrm{tt}, \quad \pi_2 \in \widehat{\mathcal{S}}^*[\![\mathsf{S}_t]\!](\pi_1\ell \xrightarrow{\mathsf{B}} \mathrm{at}[\![\mathsf{S}_t]\!])}{\ell \xrightarrow{\mathsf{B}} \mathrm{at}[\![\mathsf{S}_t]\!] \frown \pi_2 \in \widehat{\mathcal{S}}^*[\![\mathsf{S}]\!](\pi_1\ell)} \qquad (6.22)$$

- $$\frac{\mathscr{B}[\![\mathsf{B}]\!]\varrho(\pi_1\ell) = \mathrm{ff}, \quad \pi_2 \in \widehat{\mathcal{S}}^*[\![\mathsf{S}_f]\!](\pi_1\ell \xrightarrow{\neg(\mathsf{B})} \mathrm{at}[\![\mathsf{S}_f]\!])}{\ell \xrightarrow{\neg(\mathsf{B})} \mathrm{at}[\![\mathsf{S}_f]\!] \frown \pi_2 \in \widehat{\mathcal{S}}^*[\![\mathsf{S}]\!](\pi_1\ell)} \qquad (6.23)$$

A prefix finite trace of a conditional statement $\mathtt{if}\ \ell\ (\mathtt{B})\ \mathtt{S}_t\ \mathtt{else}\ \mathtt{S}_f$ continuing an initial trace $\pi_1\ell$ is the test event B (respectively $\neg$(B)) at ℓ followed by a prefix trace of S_t (respectively S_f) when boolean expression B is tt (respectively ff) on $\pi_1\ell$ in case (6.22) (respectively (6.23)).

Structural prefix trace semantics of a statement list

Prefix traces of a statement list $\texttt{Sl} ::= \texttt{Sl}'\ \texttt{S}$

- $$\frac{\pi_2 \in \widehat{\mathcal{S}}^*[\![\texttt{Sl}']\!](\pi_1)}{\pi_2 \in \widehat{\mathcal{S}}^*[\![\texttt{Sl}]\!](\pi_1)} \tag{6.13}$$

- $$\frac{\pi_2 \in \widehat{\mathcal{S}}^+[\![\texttt{Sl}']\!](\pi_1), \quad \pi_3 \in \widehat{\mathcal{S}}^*[\![\texttt{S}]\!](\pi_1 \frown \pi_2)}{\pi_2 \frown \pi_3 \in \widehat{\mathcal{S}}^*[\![\texttt{Sl}]\!](\pi_1)} \tag{6.14}$$

In case (6.14),

$$\xrightarrow{\pi_1} \underbrace{\underbrace{\begin{matrix}\text{at}[\![\texttt{Sl}]\!]\\ \text{at}[\![\texttt{Sl}']\!]\end{matrix} \xrightarrow{\pi_2} \begin{matrix}\text{after}[\![\texttt{Sl}']\!]\\ \text{at}[\![\texttt{S}]\!]\end{matrix}}_{\in\ \widehat{\mathcal{S}}^+[\![\texttt{Sl}']\!](\pi_1\text{at}[\![\texttt{Sl}']\!])} \underbrace{\xrightarrow{\pi_3} \ell}_{\in\ \widehat{\mathcal{S}}^*[\![\texttt{S}]\!](\pi_1\text{at}[\![\texttt{Sl}']\!]\pi_2\text{at}[\![\texttt{S}]\!])}}_{\in\ \widehat{\mathcal{S}}^*[\![\texttt{Sl}]\!](\pi_1\text{at}[\![\texttt{Sl}]\!])}$$

A prefix trace of $\texttt{Sl}'\ \texttt{S}$ continuing an initial trace π_1 can be a prefix trace of $\texttt{Sl}'$ or a finite maximal trace of $\texttt{Sl}'$ followed by a prefix trace of $\texttt{S}$.

Structural prefix trace semantics of an iteration statement

Prefix traces of an iteration statement $\mathtt{S} ::= \mathtt{while}\ \ell\ (\mathtt{B})\ \mathtt{S}_b$

- $$\frac{}{\ell \in \widehat{\boldsymbol{\mathcal{S}}}^*[\![\mathtt{S}]\!](\pi_1\ell)} \tag{6.24}$$

- $$\frac{\ell\pi_2\ell \in \widehat{\boldsymbol{\mathcal{S}}}^*[\![\mathtt{S}]\!](\pi_1\ell), \quad \mathscr{B}[\![\mathtt{B}]\!]\varrho(\pi_1\ell\pi_2\ell) = \mathrm{ff}}{\ell\pi_2\ell \xrightarrow{\neg(\mathtt{B})} \mathrm{after}[\![\mathtt{S}]\!] \in \widehat{\boldsymbol{\mathcal{S}}}^*[\![\mathtt{S}]\!](\pi_1\ell)} \tag{6.25}$$

- $$\frac{\begin{array}{c}\ell\pi_2\ell \in \widehat{\boldsymbol{\mathcal{S}}}^*[\![\mathtt{S}]\!](\pi_1\ell), \quad \mathscr{B}[\![\mathtt{B}]\!]\varrho(\pi_1\ell\pi_2\ell) = \mathrm{tt},\\ \pi_3 \in \widehat{\boldsymbol{\mathcal{S}}}^*[\![\mathtt{S}_b]\!](\pi_1\ell\pi_2\ell \xrightarrow{\mathtt{B}} \mathrm{at}[\![\mathtt{S}_b]\!])\end{array}}{\ell\pi_2\ell \xrightarrow{\mathtt{B}} \mathrm{at}[\![\mathtt{S}_b]\!] \frown \pi_3 \in \widehat{\boldsymbol{\mathcal{S}}}^*[\![\mathtt{S}]\!](\pi_1\ell)} \tag{6.26}$$

This is a forward, left recursive definition where $n+1$ iterations are n iterations followed by one more iteration.

Structural prefix trace semantics of an iteration statement: break statements

Remark 6.27 The inference rule (6.26) includes the case of an iteration ending with an exits by a break statement that would have the form

$$\bullet \quad \frac{\ell\pi_2\ell \in \widehat{\mathcal{S}}^*[\![\mathsf{S}]\!](\pi_1\ell), \quad \mathcal{B}[\![\mathsf{B}]\!]\varrho(\pi_1\ell\pi_2\ell) = \mathrm{tt}, \\ \pi_3 \xrightarrow{\text{break}} \text{break-to}[\![\mathsf{S}]\!] \in \widehat{\mathcal{S}}^*[\![\mathsf{S}_b]\!](\pi_1\ell\pi_2\ell \xrightarrow{\mathsf{B}} \text{at}[\![\mathsf{S}_b]\!])}{\ell\pi_2\ell \xrightarrow{\mathsf{B}} \text{at}[\![\mathsf{S}_b]\!] \frown \pi_3 \xrightarrow{\text{break}} \text{break-to}[\![\mathsf{S}]\!] \in \widehat{\mathcal{S}}^*[\![\mathsf{S}]\!](\pi_1\ell)} \tag{6.28}$$

Structural prefix trace semantics of an iteration statement

- A prefix finite trace of an iteration statement `while` ℓ (B) $\mathtt{S}_b$ continuing some initial trace $\pi_1\ell$ is
  - either ℓ (case (6.24), already covered by (6.11));
  - or, in case (6.25), the trace starting at ℓ followed by the event $\neg$(B) when the value of this boolean expression on $\pi_1\ell$ is ff and finishing at the label after⟦S⟧ after the iteration statement;
  - or, in case (6.28), the trace starting at ℓ followed by the event B when the value of this boolean expression on $\pi_1\ell$ is tt and finishing at the label at⟦$\mathtt{S}_b$⟧ followed by a prefix (indeed maximal) trace of the loop body $\mathtt{S}_b$ ending up in a break;
  - or, in case (6.26), the trace starting at ℓ, followed by a prefix trace of the iteration statement `while` ℓ (B) $\mathtt{S}_b$ representing 0 or more of iterations ending at ℓ, followed by the test event B (where the expression B is tt), followed by a prefix finite trace of the body $\mathtt{S}_t$.

Prefix trace semantics

- The prefix trace semantics is defined structurally:

$$\boldsymbol{\mathcal{S}}^{*}[\![\mathsf{S}]\!] \triangleq \widehat{\boldsymbol{\mathcal{S}}}^{*}[\![\mathsf{S}]\!]$$

- The prefix traces starting from a set $\mathscr{P}_0$ of initial traces are

$$\boldsymbol{\mathcal{S}}^{*}[\![\mathsf{S}]\!]\,\mathscr{P}_0 \triangleq \bigcup\{\boldsymbol{\mathcal{S}}^{*}[\![\mathsf{S}]\!](\pi^{\ell}) \mid \pi^{\ell} \in \mathscr{P}_0\}\,.$$

- The prefix traces starting from a set $\mathscr{P}_0$ of initial traces and arriving at program label ℓ are

$$\begin{aligned} \boldsymbol{\mathcal{S}}^{\dot{*}}[\![\mathsf{S}]\!] &\in \wp(\mathbb{T}^{+}) \stackrel{\nearrow}{\longrightarrow} (\mathbb{L} \to \wp(\mathbb{T}^{+})) \\ \boldsymbol{\mathcal{S}}^{\dot{*}}[\![\mathsf{S}]\!]\,\mathscr{P}_0\,\ell &\triangleq \{\pi_0{}^{\ell_0}\pi_1{}^{\ell_1} \mid \pi_0{}^{\ell_0} \in \mathscr{P}_0 \wedge {}^{\ell_0}\pi_1{}^{\ell_1} \in \boldsymbol{\mathcal{S}}^{*}[\![\mathsf{S}]\!](\pi_0{}^{\ell_0}) \wedge \ell_1 = \ell\} \end{aligned} \tag{6.48}$$

Example of prefix trace semantics

- $\mathtt{S} = \mathbf{while}\ \ell_1\ (\mathtt{tt})\ \ell_2\ \mathtt{x = x + 1}\ ;\ell_3.$
- $\widehat{\mathcal{S}}^*[\![\mathtt{S}]\!](\ell_1) = \left\{\left(\ell_1 \xrightarrow{\mathtt{tt}} \ell_2 \xrightarrow{\mathtt{x} = i} \ell_1\right)_{i=1}^{n}, \left(\ell_1 \xrightarrow{\mathtt{tt}} \ell_2 \xrightarrow{\mathtt{x} = i} \ell_1\right)_{i=1}^{n} \xrightarrow{\mathtt{tt}} \ell_2 \;\middle|\; n \in \mathbb{N}\right\}$ (reduced to ℓ_1 for $n = 0$).

- Notation:
  - $\left(\ell \pi(i) \ell\right)_{i=1}^{n}$ denotes the finite trace $\ell \pi(1) \ell \pi(2) \ell \ldots \pi(n) \ell$. This is the trace ℓ for $n = 0$.
  - $\left(\ell \pi(i) \ell\right)_{i=1}^{\infty}$ denotes the infinite trace $\ell \pi(1) \ell \pi(2) \ell \ldots \pi(n) \ell \pi(n+1) \ell \ldots$.

Conclusion

Conclusion

- We have defined the structural deductive stateless prefix trace semantics of a subset of C to observe partial computations of programs, where this observation can stop at any time.
- By passing to the limit, we will define the maximal trace semantics where observations terminate with the execution of the program or last for ever in case of non-termination.

The End, Thank you