Semantics of Programs
Static Analysis by Abstract Interpretation

Xavier Rival

East China Normal University

November, 27th. 2013
Choosing the right semantics is the first step in the design of a static analysis
- it should capture the relevant properties
- non relevant properties may be abstracted typically, one by one, by composing several abstractions

Abstract interpretation is a good framework to compare various semantics (independently from the application)
Application: designing lattices of semantics
Outline

1. Transition systems
2. Trace semantics
3. Denotational semantics
Programs/systems and their executions need be formalized:
- **state**: status of the machine at a given time
- **execution**: defined by transitions from a state to the next one

### Transition system (TS)

A *transition system* is a tuple \((\mathcal{S}, \rightarrow)\) where:
- \(\mathcal{S}\) is the set of states of the system
- \(\rightarrow \subseteq \mathcal{P}(\mathcal{S} \times \mathcal{S})\) is the transition relation of the system

Furthermore, transition systems may be enriched with:
- a set of initial states \(\mathcal{S}_I \subseteq \mathcal{S}\)
- a set of final states \(\mathcal{S}_F \subseteq \mathcal{S}\)

**Notes:**
- the set of states may be infinite
- steps are *discrete* (not continuous)
Program

Definition of states:

- depends on the kinds of programs to abstract
- typically, we can separate control and memory

A program is a transition system \((S, \rightarrow)\) the states of which can be described as pairs of a control state and a memory state, i.e., where:

- \(S = \mathbb{L} \times \mathbb{M}\)
- \(\mathbb{L}\) is the set of control states
- \(\mathbb{M}\) is the set of memory states

- **error state**: a distinct \(\Omega\) state, so that \(S = \mathbb{L} \times \mathbb{M} \cup \{\Omega\}\)
Transition systems

Example: imperative language

\[ i ::= x ::= e; \]
\[ \quad \text{if}(c) \ b \ \text{else} \ b \]
\[ \quad \text{while}(c) \ b \]

\[ b ::= \{ i; \ldots ; i; \} \]

- \( X \): finite, predefined set of variables
- \( L \): before and after each statement

Definition of \( \rightarrow \):

transitions for all instructions

- \( l_0 : x = e; \ l_1 : \)
  - if \( \llbracket e \rrbracket(m) \neq \Omega \), then
  \( (l_0, m) \rightarrow (l_1, m[x \leftarrow \llbracket e \rrbracket(\rho)]) \)
  - if \( \llbracket e \rrbracket(m) = \Omega \), then
  \( (l_0, m) \rightarrow \Omega \)

- \( l_0 : \text{while}(c)\{ l_1 : b \_t \; l_2 \} \; l_3 : \)
  - if \( \llbracket e \rrbracket(m) = \text{true} \), then
  \( (l_0, m) \rightarrow (l_1, m) \)
  \( (l_2, m) \rightarrow (l_1, m) \)
  - if \( \llbracket e \rrbracket(m) = \text{false} \), then
  \( (l_0, m) \rightarrow (l_3, m) \)
  \( (l_2, m) \rightarrow (l_3, m) \)
  - if \( \llbracket e \rrbracket(m) = \Omega \), then
  \( (l_0, m) \rightarrow \Omega \)
  \( (l_2, m) \rightarrow \Omega \)
Outline

1 Transition systems

2 Trace semantics
   - Finite traces
   - Infinite traces
   - Finite and infinite traces
   - Abstraction relations

3 Denotational semantics
Traces: definitions

- A trace is a finite or infinite sequence of states.

Notations

- We write $\langle s_0, \ldots, s_n \rangle$ for a finite trace and $\langle s_0, \ldots \rangle$ for an infinite trace.
- $S^*$ is the set of finite traces.
- $S^\omega$ is the set of infinite traces.
- $S^{bif} = S^* \cup S^\omega$ is the set of finite or infinite traces.
Trace semantics

Operations on traces

- **length** $|\sigma|:$
  \[
  \begin{cases}
  \langle s_0, \ldots, s_n \rangle = n + 1 \\
  \langle s_0, \ldots \rangle = \omega
  \end{cases}
  \]

- **prefix** order relation:
  \[
  \langle s_0, \ldots, s_n \rangle \prec \langle s'_0, \ldots, s'_{n'} \rangle \iff \begin{cases}
    n \leq n' \\
    \forall i \in [0, n], s_i = s'_i
  \end{cases}
  \]

(also defined for infinite traces)

- **concatenation** operator "·":
  \[
  \begin{align*}
  \langle s_0, \ldots, s_n \rangle \cdot \langle s'_0, \ldots, s'_{n'} \rangle &= \langle s_0, \ldots, s_n, s'_0, \ldots, s'_{n'} \rangle \\
  \langle s_0, \ldots, s_n \rangle \cdot \langle s'_0, \ldots \rangle &= \langle s_0, \ldots, s_n, s'_0, \ldots \rangle \\
  \langle s_0, \ldots, s_n, \ldots \rangle \cdot \sigma' &= \langle s_0, \ldots, s_n, \ldots \rangle
  \end{align*}
  \]

- **empty trace** $\epsilon$, neutral element for ·.
Semantics of finite traces

Goal: capture all finite executions of the program

We consider a transition system $S = (S, \rightarrow)$

**Definition**

The **finite traces semantics** $[S]^*$ is defined by:

$$[S]^* = \{ \langle s_0, \ldots, s_n \rangle \in S^* \mid \forall i, s_i \rightarrow s_{i+1} \}$$

**Example:**

- contrived transition system $S = (\{a, b, c, d\}, \{(a, b), (b, a), (b, c)\})$
- finite traces semantics:

$$[S]^* = \{ \langle a, b, \ldots, a, b, a \rangle, \langle b, a, \ldots, a, b, a \rangle, \langle a, b, \ldots, a, b, a, b \rangle, \langle b, a, \ldots, a, b, a, b \rangle, \langle a, b, \ldots, a, b, a, b, c \rangle, \langle b, a, \ldots, a, b, a, b, c \rangle, \langle c \rangle, \langle d \rangle \}$$
Interesting subsets of the finite trace semantics

We consider a transition system $S = (S, \rightarrow, S_I, S_F)$

- the traces from an initial state:
  \[
  \{\langle s_0, \ldots, s_n \rangle \in [S]^* \mid s_0 \in S_I\}
  \]

- the traces reaching a blocking state:
  \[
  \{\sigma \in [S]^* \mid \forall \sigma' \in [S]^*, \sigma \prec \sigma' \implies \sigma = \sigma'\}
  \]

- the traces ending in a final state:
  \[
  \{\langle s_0, \ldots, s_n \rangle \in [S]^* \mid s_n \in S_F\}
  \]

Example (same transition system, with $S_I = \{a\}$ and $S_F = \{c\}$):

- traces from an initial state ending in a final state:
  \[
  \{\langle a, b, \ldots, a, b, a, b, c \rangle\}
  \]
Fixpoint definition for the semantics of finite traces

We consider a transition system \( S = (S, \rightarrow) \).
The semantics of finite traces can be defined as a least-fixpoint:

**Finite traces semantics as a fixpoint**

Let \( \mathcal{I} = \{ \langle s \rangle \mid s \in S \} \). Let \( F_* \) be the function defined by:

\[
F_* : \mathcal{P}(S^*) \longrightarrow \mathcal{P}(S^*)
\]

\[
X \longmapsto X \cup \{ \langle s_0, \ldots, s_n, s_{n+1} \rangle \mid \langle s_0, \ldots, s_n \rangle \in X \land s_n \rightarrow s_{n+1} \}
\]

Then, \( F_* \) is continuous and thus has a least-fixpoint greater than \( \mathcal{I} \); moreover:

\[
\text{Ifp}_\mathcal{I} F_* = [S]^* = \bigcup_{n \in \mathbb{N}} F_*^n(\mathcal{I})
\]
Trace semantics  Finite traces

Fixpoint definition: proof (1), fixpoint existence

First, we prove that $F_\star$ is continuous. Let $\mathcal{X} \subseteq \mathcal{P}(S^\star)$ and $A = \bigcup_{X \in \mathcal{X}} X$. Then:

$$F_\star(\bigcup_{X \in \mathcal{X}} X)$$

$$= A \cup \{\langle s_0, \ldots, s_n, s_{n+1} \rangle \mid (\langle s_0, \ldots, s_n \rangle \in \bigcup_{X \in \mathcal{X}} X) \land s_n \rightarrow s_{n+1}\}$$

$$= A \cup \{\langle s_0, \ldots, s_n, s_{n+1} \rangle \mid (\exists X \in \mathcal{X}, \langle s_0, \ldots, s_n \rangle \in X) \land s_n \rightarrow s_{n+1}\}$$

$$= A \cup \{\langle s_0, \ldots, s_n, s_{n+1} \rangle \mid \exists X \in \mathcal{X}, \langle s_0, \ldots, s_n \rangle \in X \land s_n \rightarrow s_{n+1}\}$$

$$= (\bigcup_{X \in \mathcal{X}} X) \cup (\bigcup_{X \in \mathcal{X}} \{\langle s_0, \ldots, s_n, s_{n+1} \rangle \mid \langle s_0, \ldots, s_n \rangle \in X \land s_n \rightarrow s_{n+1}\})$$

$$= \bigcup_{X \in \mathcal{X}} (X \cup \{\langle s_0, \ldots, s_n, s_{n+1} \rangle \mid \langle s_0, \ldots, s_n \rangle \in X \land s_n \rightarrow s_{n+1}\})$$

$$= \bigcup_{X \in \mathcal{X}} F_\star(X)$$

Function $F_\star$ is $\cup$-complete, hence continuous.

As $(\mathcal{P}(S^\star), \subseteq)$ is a CPO, the continuity of $F_\star$ entails the existence of a least-fixpoint (Kleene theorem); moreover, it implies that:

$$\text{lfp}_I F_\star = \bigcup_{n \in \mathbb{N}} F_\star^n(I)$$
Fixpoint definition: proof (2), fixpoint equality

We now show that $[S]^*$ is equal to $\text{Ifp}_\mathcal{I} F_\ast$, by showing the property below, by induction over $n$:

$$\forall k \leq n, \langle s_0, \ldots, s_n \rangle \in F_\ast^n(\mathcal{I}) \iff \langle s_0, \ldots, s_n \rangle \in [S]^*$$

- at rank 0, only traces of length 1 need be considered:

  $$\langle s \rangle \in [S]^* \iff s \in S \iff \langle s \rangle \in F_\ast^0(\mathcal{I})$$

- at rank $n + 1$, and assuming the property holds at rank $n$ (the equivalence is obvious for traces of length 1):

  $$\langle s_0, \ldots, s_k, s_{k+1} \rangle \in [S]^*$$
  $$\iff \langle s_0, \ldots, s_k \rangle \in [S]^* \land s_k \rightarrow s_{k+1}$$
  $$\iff \langle s_0, \ldots, s_k \rangle \in F_\ast^n(\mathcal{I}) \land s_k \rightarrow s_{k+1} \quad (k \leq n \text{ since } k + 1 \leq n + 1)$$
  $$\iff \langle s_0, \ldots, s_k, s_{k+1} \rangle \in F_\ast^{n+1}(\mathcal{I})$$
Example, with the same simple transition system $S = (\mathbb{S}, \rightarrow)$:

- $\mathbb{S} = \{a, b, c, d\}$
- $\rightarrow$ is defined by $a \rightarrow b$, $b \rightarrow a$ and $b \rightarrow c$

Then, the first iterates are:

$$
F^0_\star(\mathcal{I}) = \{\langle a \rangle, \langle b \rangle, \langle c \rangle, \langle d \rangle\}
$$

$$
F^1_\star(\mathcal{I}) = F^0_\star(\mathcal{I}) \cup \{\langle b, a \rangle, \langle a, b \rangle, \langle b, c \rangle\}
$$

$$
F^2_\star(\mathcal{I}) = F^1_\star(\mathcal{I}) \cup \{\langle a, b, a \rangle, \langle b, a, b \rangle, \langle a, b, c \rangle\}
$$

$$
F^3_\star(\mathcal{I}) = F^2_\star(\mathcal{I}) \cup \{\langle b, a, b, a \rangle, \langle a, b, a, b \rangle, \langle b, a, b, c \rangle\}
$$

$$
F^4_\star(\mathcal{I}) = F^3_\star(\mathcal{I}) \cup \{\langle a, b, a, b, a \rangle, \langle b, a, b, a, b \rangle, \langle a, b, a, b, c \rangle\}
$$

$$
F^5_\star(\mathcal{I}) = \ldots
$$

- the traces of $\llbracket S \rrbracket^\star$ of length $n + 1$ appear in $F^n_\star(\mathcal{I})$
So far, we do not really isolate non-terminating behaviors

We consider a transition system $S = (S, \rightarrow)$

**Definition**

The **infinite traces semantics** $[S]^{\omega}$ is defined by:

$$[S]^{\omega} = \{ \langle s_0, \ldots \rangle \in S^{\omega} \mid \forall i, s_i \rightarrow s_{i+1} \}$$

**Example:**

- contrived transition system defined by
  $$S = \{ a, b, c, d \} \quad \text{(→)} = \{ (a, b), (b, a), (b, c) \}$$
- the infinite traces semantics contains only two traces
  $$[S]^{\omega} = \{ \langle a, b, \ldots, a, b, a, b, \ldots \rangle, \langle b, a, \ldots, b, a, b, a, \ldots \rangle \}$$
Semantics of infinite traces: towards a fixpoint form

Can we also provide a fixpoint form for $[S]^\omega$?

Intuitively, $\langle s_0, s_1, \ldots \rangle \in [S]^\omega$ if and only if $\forall n$, $s_n \rightarrow s_{n+1}$, i.e.,

$$\forall n \in \mathbb{N}, \forall k \leq n, s_k \rightarrow s_{k+1}$$

Let $F_\omega$ be defined by:

$$F_\omega : \mathcal{P}(S^\omega) \rightarrow \mathcal{P}(S^\omega)$$

$$X \mapsto \{\langle s_0, s_1, \ldots, s_n, \ldots \rangle | \langle s_1, \ldots, s_n, \ldots \rangle \in X \land s_0 \rightarrow s_1 \}$$

Then, we can show by induction that:

$$\sigma \in [S]^\omega \iff \forall n \in \mathbb{N}, \sigma \in F_\omega^n(S^\omega) \iff \bigcap_{n \in \mathbb{N}} F_\omega^n(S^\omega)$$
### Duality principle

- If $\subseteq$ is an order relation, so is $\supseteq$
- All properties of $\subseteq$ are inherited by $\supseteq$, modulo some correspondance

<table>
<thead>
<tr>
<th>Basic order</th>
<th>Dual order</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\subseteq$</td>
<td>$\supseteq$</td>
</tr>
<tr>
<td>$\cup$</td>
<td>$\cap$</td>
</tr>
<tr>
<td>$\cap$</td>
<td>$\cup$</td>
</tr>
<tr>
<td>$\bot$</td>
<td>$\top$</td>
</tr>
<tr>
<td>$\cup$-continuous function</td>
<td>$\cap$-continuous function</td>
</tr>
<tr>
<td>$\cap$-continuous function</td>
<td>$\cup$-continuous function</td>
</tr>
<tr>
<td>Least-fixpoint (lfp)</td>
<td>Greatest-fixpoint (gfp)</td>
</tr>
<tr>
<td>Greatest-fixpoint (gfp)</td>
<td>Least-fixpoint (lfp)</td>
</tr>
</tbody>
</table>

Thus, we can derive dual versions of Tarski’s theorem and Kleene’s theorem.
Fixpoint form of the semantics of infinite traces

Infinite traces semantics as a fixpoint

Let $F_\omega$ be the function defined by:

$$
F_\omega : \mathcal{P}(S^\omega) \rightarrow \mathcal{P}(S^\omega)
$$

$$
X \mapsto \{ \langle s_0, s_1, \ldots, s_n, \ldots \rangle \mid \langle s_1, \ldots, s_n, \ldots \rangle \in X \land s_0 \rightarrow s_1 \}
$$

Then, $F_\omega$ is $\cap$-continuous and thus has a greatest-fixpoint; moreover:

$$
gfp_{S^\omega} F_\omega = \llbracket S \rrbracket^\omega = \bigcap_{n \in \mathbb{N}} F^n_\omega(S^\omega)
$$

Proof sketch:

- the $\cap$-continuity proof is similar as for the $\cup$-continuity of $F_*$
- by the dual version of Kleene’s theorem, $gfp_{S^\omega} F_\omega$ exists and is equal to $\bigcap_{n \in \mathbb{N}} F^n_\omega(S^\omega)$, i.e. to $\llbracket S \rrbracket^\omega$ (induction proof)
Example

Example, with the same simple transition system:

- $S = \{a, b, c, d\}$
- $\rightarrow$ is defined by $a \rightarrow b$, $b \rightarrow a$ and $b \rightarrow c$

Then, the first iterates are:

\[
\begin{align*}
F^0_\omega(S^\omega) &= S^\omega \\
F^1_\omega(S^\omega) &= \langle a, b \rangle \cdot S^\omega \cup \langle b, a \rangle \cdot S^\omega \cup \langle b, c \rangle \cdot S^\omega \\
F^2_\omega(S^\omega) &= \langle b, a, b \rangle \cdot S^\omega \cup \langle a, b, a \rangle \cdot S^\omega \cup \langle a, b, c \rangle \cdot S^\omega \\
F^3_\omega(S^\omega) &= \langle a, b, a, b \rangle \cdot S^\omega \cup \langle b, a, b, a \rangle \cdot S^\omega \cup \langle b, a, b, c \rangle \cdot S^\omega \\
F^4_\omega(S^\omega) &= \ldots
\end{align*}
\]

Intuition

- at iterate $n$, prefixes of length $n + 1$ match the traces in the infinite semantics
- only $\langle a, b, \ldots, a, b, a, b, \ldots \rangle$ and $\langle b, a, \ldots, b, a, b, a, \ldots \rangle$ belong to all iterates
Maximal traces semantics

The maximal traces semantics simply puts together the finite traces semantics and the infinite traces semantics:

We consider a transition system $S = (S, \rightarrow)$

**Definition**

The **maximal traces semantics** $[S]^{\text{bif}}$ is the element of $S^{\text{bif}}$ defined by:

$$[S]^{\text{bif}} = [S]^* \cup [S]^\omega$$
Example

Still same simple transition system:

- $S = \{a, b, c, d\}$
- $\rightarrow$ is defined by $a \rightarrow b$, $b \rightarrow a$ and $b \rightarrow c$

Then:

$$[S]^{\text{bif}} = \{ \langle a, b, \ldots, a, b, a \rangle, \langle b, a, \ldots, a, b, a \rangle, \langle a, b, \ldots, a, b, a, b \rangle, \langle b, a, \ldots, a, b, a, b \rangle, \langle a, b, \ldots, a, b, a, b, c \rangle, \langle b, a, \ldots, a, b, a, b, c \rangle, \langle c \rangle, \langle d \rangle, \langle a, b, \ldots, a, b, a, b, \ldots \rangle, \langle b, a, \ldots, b, a, b, a, \ldots \rangle \}$$
Co-induction technique

Goal of the co-induction technique

- how to set up a new fixpoint definition?
- we need to combine a least-fixpoint and a greatest-fixpoint

- **lattice**: $\mathcal{S}^{\text{bif}}$, with the order relation $\sqsubseteq^{\text{bif}}$ defined by

\[
X \sqsubseteq^{\text{bif}} Y \iff \{ X \cap S^* \subseteq Y \cap S^* \land X \cap S^\omega \supseteq Y \cap S^\omega \}
\]

- **join**: $X \cup Y = ((X \cap S^*) \cup (Y \cap S^*)) \cup ((X \cap S^\omega) \cap (Y \cap S^\omega))$

- **assumptions**: we assume $F^*$ and $F^\omega$ defined as before

- **semantic function** $F_{\text{bif}}$ defined by:

\[
F_{\text{bif}} : \mathcal{P}(\mathcal{S}^{\text{bif}}) \rightarrow \mathcal{P}(\mathcal{S}^{\text{bif}})
\]

\[
X \mapsto F^*(X \cap S^*) \cup F^\omega(X \cap S^\omega)
\]
Fixpoint form of the maximal trace semantics

We have the following properties:

- \((\mathcal{S}_{\text{bif}}, \sqsubseteq_{\text{bif}}, \sqcup_{\text{bif}})\) is a complete lattice
- \(F_{\text{bif}}\) is \(\sqcup_{\text{bif}}\)-continuous
- thus, it has a least-fixpoint greater than \(\mathcal{I} = \{\langle s \rangle \mid s \in \mathcal{S}\}\); furthermore:

\[
\begin{align*}
\text{lfp}_\mathcal{I} F_{\text{bif}} \cap \mathcal{S}^* & = \text{lfp}_\mathcal{I} F_* \\
\text{lfp}_\mathcal{I} F_{\text{bif}} \cap \mathcal{S}^{\omega} & = \text{gfp}_{\mathcal{S}^{\omega}} F^{\omega} \\
\text{lfp}_\mathcal{I} F_{\text{bif}} & = \text{lfp}_\mathcal{I} F_* \cup \text{gfp}_{\mathcal{S}^{\omega}} F^{\omega}
\end{align*}
\]

Therefore:

Fixpoint definition of \([\mathcal{S}]_{\text{bif}}\)

\[ [\mathcal{S}]_{\text{bif}} = \text{lfp}_\mathcal{I} F_{\text{bif}} \]
Finite traces as an abstraction

- We have defined three semantics; how to relate them? Can this be done in a constructive manner?
- Abstract interpretation allows to define relation between semantics!

The finite semantics discards the infinite executions

Finite traces abstraction

We define $\alpha_\ast$, $\gamma_\ast$ by:

$$
\alpha_\ast : \mathcal{P}(S^{\text{bif}}) \rightarrow \mathcal{P}(S^\ast)
$$

$X \mapsto X \cap S^\ast$

$$
\gamma_\ast : \mathcal{P}(S^\ast) \rightarrow \mathcal{P}(S^{\text{bif}})
$$

$Y \mapsto Y \cup S^\omega$

Then:

- These define a Galois connection $(\mathcal{P}(S^{\text{bif}}), \subseteq) \xleftarrow{\gamma_\ast} (\mathcal{P}(S^\ast), \subseteq)$

- Moreover, $\alpha_\ast(\llbracket S \rrbracket^{\text{bif}}) = \llbracket S \rrbracket^\ast$

Proof: $\forall X \in \mathcal{P}(S^{\text{bif}}), Y \in \mathcal{P}(S^\ast), \alpha_\ast(X) \subseteq Y \iff X \subseteq \gamma_\ast(Y)$
Fixpoint transfer

We can actually make this statement more constructive

**Exact fixpoint transfer**

Let \((\mathcal{D}_0, \sqsubseteq_0)\) and \((\mathcal{D}_1, \sqsubseteq_1)\) be two domains, let \(\alpha, \gamma\) be a pair of adjoint functions defining a Galois connection \((\mathcal{D}_0, \sqsubseteq_0) \xleftarrow{\alpha} \xrightarrow{\gamma} (\mathcal{D}_1, \sqsubseteq_1)\).

Let \(F_0 : \mathcal{D}_0 \to \mathcal{D}_0\), \(F_1 : \mathcal{D}_1 \to \mathcal{D}_1\) and \(x_0 \in \mathcal{D}_0, x_1 \in \mathcal{D}_1\), such that:

- \(F_0\) is continuous
- \(F_1\) is monotone
- \(\alpha \circ F_0 = F_1 \circ \alpha\)
- \(\alpha(x_0) = x_1\)

Then:

- both \(F_0\) and \(F_1\) have a least-fixpoint (Tarski’s fixpoint theorem)
- \(\alpha(lfp_{x_0} F_0) = lfp_{x_1} F_1\)
Fixpoint transfer: proof

- $\alpha(lfpF_0)$ is a fixpoint of $F_1$ since:

\[
F_1(\alpha(lfp_{x_0} F_0)) = \alpha(F_0(lfp_{x_0} F_0)) = \alpha(lfp_{x_0} F_0)
\]

since $\alpha \circ F_0 = F_1 \circ \alpha$ by definition of the fixpoints

- to show that $\alpha(lfp_{x_0} F_0)$ is the least-fixpoint of $F_1$, we assume that $X$ is another fixpoint of $F_1$ and we show that $\alpha(lfp_{x_0} F_0) \subseteq_1 X$, i.e., that $lfp_{x_0} F_0 \subseteq_0 \gamma(X)$; as $lfp_{x_0} F_0 = \bigcup_{n \in \mathbb{N}} F_0^n(x_0)$, it amounts to proving that $\forall n \in \mathbb{N}, F_0^n(x_0) \subseteq_0 \gamma(X)$;

by induction over $n$:

- $F_0^0(x_0) = x_0$, thus $\alpha(F_0^0(x_0)) = x_1 \subseteq_0 \gamma(X)$;

- let us assume that $F_0^n(x_0) \subseteq_0 \gamma(X)$, and let us show that $F_0^{n+1}(x_0) \subseteq_0 \gamma(X)$, i.e. that $\alpha(F_0^{n+1}(x_0)) \subseteq_1 X$:

\[
\alpha(F_0^{n+1}(x_0)) = \alpha \circ F_0(F_0^n(x_0)) = F_1 \circ \alpha(F_0^n(x_0)) \subseteq_1 F_1(X) = X
\]

as $\alpha(F_0^n(x_0)) \subseteq_1 X$
Application of the fixpoint transfer

All assumptions are satisfied:

- $\alpha_\star, \gamma_\star$ define a Galois connection between $(\mathcal{P}(S^{\text{bif}}), \subseteq)$ and $(\mathcal{P}(S^*), \subseteq)$
- $\alpha_\star(\mathcal{I}) = \mathcal{I}$
- $F_{\text{bif}}$ is continuous
- $F_\star$ is continuous, hence monotone
- $F_\star \circ \alpha_\star = \alpha_\star \circ F_{\text{bif}}$

This gives another proof of the abstraction relation:

\[
\alpha_\star([S]^{\text{bif}}) = \alpha_\star(\text{lfp}_\mathcal{I} F_{\text{bif}}) = \text{lfp}_\mathcal{I} F_\star = [S]^*
\]

The constructive proof ties very closely the iterates i.e., the way the semantics are computed
Infinite traces as an abstraction

The same reasoning can be applied to the infinite traces semantics:

Infinite traces abstraction

We define $\alpha_\omega, \gamma_\omega$ by:

$\alpha_\omega : \mathcal{P}(S^{\text{bif}}) \longrightarrow \mathcal{P}(S^\omega)$ \hspace{1cm} $\gamma_\omega : \mathcal{P}(S^\omega) \longrightarrow \mathcal{P}(S^{\text{bif}})$

$X \longmapsto X \cap S^\omega$ \hspace{1cm} $Y \longmapsto Y \cup S^*$

Then:

- these define a Galois connection $(\mathcal{P}(S^{\text{bif}}), \subseteq) \xleftarrow{\gamma_\omega} (\mathcal{P}(S^\omega), \subseteq)$ \xrightarrow{\alpha_\omega} $(\mathcal{P}(S^{\text{bif}}), \subseteq)$

- moreover, $\alpha_\omega([S]^{\text{bif}}) = [S]^\omega$

- the fixpoint transfer also holds: $\alpha_\omega \circ F_{\text{bif}} = F_\omega \circ \alpha_\omega$, $F_{\text{bif}}$ is continuous and $F_\omega$ is continuous, hence monotone
Towards a hierarchy of semantics

So far, we have:

- three forms of operational semantics
- two abstraction relations

\[
\begin{align*}
[S]^* & \quad \alpha_* \quad [S]_{\text{bif}} \quad \alpha_\omega \\
[S]_{\omega} & \quad \alpha_* \\
[S] & \quad \alpha_\omega
\end{align*}
\]

We can actually build lattices of semantics:
“greater” means “more abstract than”

*Constructive Design of a Hierarchy of Semantics of a Transition System by Abstract Interpretation.*

Patrick Cousot.

In *Electronic Notes in Theoretical Computer Science, 6* (1997)
Outline

1 Transition systems

2 Trace semantics

3 Denotational semantics
   - Denotational semantics and finite behaviors
   - Reachable states
Denotational semantics: definition

- The operational (trace) semantics is very precise: it records \textit{all the history} of all executions of the system.
- This may be too precise in many cases, e.g., when the history is not relevant.

- We first focus on the \textit{finite} behaviors.
- We consider transition system \( S = (\mathcal{S}, \rightarrow) \).

Finite denotational semantics

The denotational semantics \([S]_\partial\) is the function

\[
[S]_\partial : \mathcal{S} \longrightarrow \mathcal{P}(\mathcal{S})
\]

\[
s \longmapsto \{s' \in \mathcal{S} \mid s \rightarrow^* s'\}
\]

Semantic domain: \( \mathcal{D}_\partial = \mathcal{S} \rightarrow \mathcal{P}(\mathcal{S}) \), with the pointwise extension of \( \subseteq \).
Example

Another contrived transition system $S = (\mathbb{S}, \rightarrow)$ defined by:

- $\mathbb{S} = \{a, b, c, d\}$
- $a \rightarrow b$, $c \rightarrow c$, $c \rightarrow d$

Then:

$$[S]_{\partial} : \begin{align*}
a & \mapsto \{a, b\} \\
b & \mapsto \{b\} \\
c & \mapsto \{c, d\} \\
d & \mapsto \{d\}
\end{align*}$$

Observations

- much more compact than the operational semantics
- the execution history is effectively left behind
- the semantics makes no difference between one step and a sequence of any number of steps (as observed from state $c$)
Denotational abstraction

We can obviously derive $[S]_{\partial}$ from $[S]^*$

Definition of the denotational abstraction

Let $\alpha_{\partial}, \gamma_{\partial}$ be the functions defined by

$$
\alpha_{\partial} : \mathcal{P}(S^*) \longrightarrow D_{\partial} \\
X \quad \longmapsto \quad \lambda s_0 \cdot \{ s_n \in S \mid \exists \sigma = \langle s_0, \ldots, s_n \rangle \in X \}
$$

$$
\gamma_{\partial} : D_{\partial} \longrightarrow \mathcal{P}(S^*) \\
\Psi \quad \longmapsto \quad \{ \langle s_0, \ldots, s_n \rangle \in S^* \mid s_n \in \Psi(s_0) \}
$$

These functions form a Galois connection

$$(\mathcal{P}(S^*), \subseteq) \xleftarrow{\gamma_{\partial}} (D_{\partial}, \subseteq) \xrightarrow{\alpha_{\partial}} (\mathcal{P}(S^*), \subseteq)$$

Proof: straightforward computation
Abstraction relation

Following the definitions of $[.]\partial$, $[.]^*$ and $\alpha\partial$:

$$[S]_\partial = \alpha\partial([S]^*)$$

Other similar kinds of abstractions:

- Relational semantics
- Pre-conditions (e.g., weakest pre-conditions semantics)
Fixpoint definition

Can \([S]_\partial\) be constructively defined? Yes, fixpoint transfer!

With the notations used so far for \(S\), its semantics and semantic functions, and with \(X \in \mathcal{P}(S^*)\),

\[
\alpha_\partial \circ F_*(X) = \lambda(s \in S) \cdot \{s' \in S \mid \exists\langle s, \ldots, s'\rangle \in F_*(X)\} \\
= \lambda(s_0 \in S) \cdot \{s_{n+1} \in S \mid \exists\langle s_0, \ldots, s_n\rangle \in X \land s_n \rightarrow s_{n+1}\} \\
= \lambda(s_0 \in S) \cdot \{s_{n+1} \in S \mid \exists s_n \in \alpha_\partial(X), s_n \rightarrow s_{n+1}\} \\
= F_\partial \circ \alpha_\partial(X)
\]

where:

\[
F_\partial : \mathcal{D}_\partial \rightarrow \mathcal{D}_\partial \\
\psi \mapsto \lambda(s \in S) \cdot \{s' \in S \mid s \rightarrow s'\}
\]
Fixpoint form of the denotational semantics

We remark that:

- \((\mathcal{P}(\mathcal{S}^*), \subseteq)\) and \((\mathbb{D}_\partial, \subseteq)\) are complete lattices
- \(\alpha_\partial, \gamma_\partial\) define a Galois connection between these lattices
- \(F_\ast\) is continuous
- \(F_\partial\) is continuous, hence monotone
- \(\alpha_\partial \circ F_\ast = F_\partial \circ \alpha_\partial\)
- \(\alpha_\partial(\mathcal{I}) = \alpha_\partial(\{\langle s \rangle \mid s \in \mathcal{S}\}) = \lambda(s \in \mathcal{S}) \cdot \{s\}\)
  
  (we write \(\mathcal{I}\) for the identity function)

Therefore, by fixpoint transfer:

\[
\llbracket \mathcal{S} \rrbracket_\partial = \alpha_\partial(\llbracket \mathcal{S} \rrbracket^*) = \alpha_\partial(\text{lfp}_\mathcal{I} F_\ast) = \text{lfp}_\mathcal{I} F_\partial
\]
Applications

The choice of the concrete semantics is tied to the properties to analyze.

Denotational semantics is a good basis for:
- modular analyses, based on the abstraction of input-output relations
- typing analyses: types are an abstraction of the denotational semantics
- whenever intermediate states are not relevant, it is helpful to abstract them
Reachable states abstraction

We consider a transition system $\mathcal{S} = (\mathbb{S}, \rightarrow, \mathbb{S}_I)$

**Definition**

We let $\alpha_R$ be defined by:

$$\alpha_R : \mathbb{D}_\emptyset \longrightarrow \mathcal{P}(\mathbb{S})$$

$$\Phi \longmapsto \Phi(\mathbb{S}_I)$$

$$\gamma_R : \mathcal{P}(\mathbb{S}) \longrightarrow \mathbb{D}_\emptyset$$

$$\chi \longmapsto \lambda(s \in \mathbb{S}) \cdot \begin{cases} X & \text{if } s \in \mathbb{S}_I \\ \mathbb{S} & \text{otherwise} \end{cases}$$

Then, we have a Galois connection $(\mathbb{D}_\emptyset, \subseteq) \xrightarrow{\gamma_R} (\mathcal{P}(\mathbb{S}), \subseteq)$. We let:

$$[\mathcal{S}]_R = \alpha_R([\mathcal{S}]_\emptyset) = \{ s_n \in \mathbb{S} \mid \exists \langle s_0, \ldots, s_n \rangle \in [\mathcal{S}]_R, \ s_0 \in \mathbb{S}_I \}$$
Example, with the simple transition system $S$ defined by:

- $S = \{a, b, c, d\}$
- $\rightarrow$ is defined by $a \rightarrow b$, $b \rightarrow a$ and $b \rightarrow c$
- $S_I = \{a\}$

Then, the **reachable states** are:

$$[S]_R = \{a, b, c\}$$
Composition of Galois connections

Composition property

Let \((\mathbb{D}_0, \sqsubseteq_0), (\mathbb{D}_1, \sqsubseteq_1)\) and \((\mathbb{D}_2, \sqsubseteq_2)\) be three abstract domains, and let us assume the Galois connections below are defined:

\[
(\mathbb{D}_0, \sqsubseteq_0) \xleftarrow{\alpha_{01}} \xrightarrow{\gamma_{10}} (\mathbb{D}_1, \sqsubseteq_1) \quad (\mathbb{D}_1, \sqsubseteq_1) \xleftarrow{\alpha_{12}} \xrightarrow{\gamma_{21}} (\mathbb{D}_2, \sqsubseteq_2)
\]

Then, we have a third Galois connection

\[
(\mathbb{D}_0, \sqsubseteq_0) \xleftarrow{\alpha_{12} \circ \alpha_{01}} \xrightarrow{\gamma_{10} \circ \gamma_{21}} (\mathbb{D}_2, \sqsubseteq_2)
\]

Proof: if \(x_0 \in \mathbb{D}_0, x_2 \in \mathbb{D}_2\), then

\[
\alpha_{12} \circ \alpha_{01}(x_0) \sqsubseteq_2 x_2 \iff \alpha_{01}(x_0) \sqsubseteq_1 \gamma_{21}(x_2) \iff x_0 \sqsubseteq_0 \gamma_{10} \circ \gamma_{21}(x_2)
\]

Application

\([S]_\mathcal{R}\) is also an abstraction of \([S]^*\)
Fixpoint form of the reachable states abstraction

Fixpoint definition

We let $F_R$ be defined by:

$$F_R : \mathcal{P}(S) \longrightarrow \mathcal{P}(S)$$

$$X \longmapsto \{ s \in S | \exists s' \in X, s' \rightarrow s \}$$

Then, $F_R$ is continuous, has a least fixpoint and

$$[S]_R = \text{lfp}_{S,I} F_R$$

Proof: fixpoint transfer
Lattice of abstractions

- Abstraction is a **pre-order relation** among semantics.
- These semantics can be compared by abstraction.
- They form a **lattice** of semantics.
- We can define and compare more semantics.