Contents & Goals

Last Lecture:
- System configuration cont’d
- Action language and transformer

This Lecture:
- Educational Objectives: Capabilities for following tasks/questions.
  - What does this State Machine mean? What happens if I inject this event?
  - Can you please model the following behaviour.
  - What is: Signal, Event, Ether, Transformer, Step, RTC.

- Content:
  - Step, Run-to-Completion Step
Transition Relation

**Definition.** Let $A$ be a set of labels and $S$ a (not necessarily finite) set of states. We call

$$\rightarrow \subseteq S \times A \times S$$

a (labelled) transition relation.

Let $S_0 \subseteq S$ be a set of initial states. A (finite or infinite) sequence

$${s_0} \xrightarrow{a_0} s_1 \xrightarrow{a_1} s_2 \xrightarrow{a_2} \ldots$$

with $s_i \in S$, $a_i \in A$ is called computation of the labelled transition system $(S, A, \rightarrow, S_0)$ if and only if

- **initiation**: $s_0 \in S_0$
- **consecution**: $(s_i, a_i, s_{i+1}) \in \rightarrow$ for $i \in \mathbb{N}_0$. 
**Active vs. Passive Classes/Objects**

- **Note:** From now on, for simplicity, assume that all classes are **active**.

  We’ll later briefly discuss the Rhapsody framework which proposes a way how to integrate non-active objects.

- **Note:** The following RTC “algorithm” follows Harel and Gery (1997) (i.e. the one realised by the Rhapsody code generation) if the standard is ambiguous or leaves choices.

---

**From Core State Machines to LTS**

**Definition.** Let $\mathcal{S}_0 = (\mathcal{F}_0, \mathcal{V}_0, \mathcal{V}_0, \mathcal{A}_0, \mathcal{E})$ be a signature with signals (all classes in $\mathcal{C}_0$ **active**), $\mathcal{D}_0$ a structure of $\mathcal{S}_0$, and $(\mathcal{E}, \text{ready}, \oplus, \ominus, \cdot)$ an ether over $\mathcal{S}_0$ and $\mathcal{D}_0$. Assume there is one core state machine $M_C$ per class $C \in \mathcal{C}$.

We say, the state machines **induce** the following labelled transition relation on states $S := (\Sigma \times \mathcal{E}) \cup \{\#\}$ with labels $A := 2^\mathcal{E} \times 2^{(\mathcal{E} \cup \{+, \cdot\})} \times 2^{\mathcal{C}}$.

- $\bullet (\sigma, \epsilon) (\text{cons, Snd}) (\sigma', \epsilon')$ if and only if
  (i) an event with destination $u$ is **discarded**,
  (ii) an event is **dispatched** to $u$, i.e. stable object processes an event, or
  (iii) run-to-completion processing by $u$ **continues**, i.e. object $u$ is not stable and continues to process an event,
  (iv) the environment interacts with object $u$,

- $\bullet s (\text{cons}, \emptyset) \xrightarrow{u} \#$ if and only if
  (v) an **error condition** occurs during consumption of cons, or
  $s = \#$ and $\text{cons} = \emptyset$. 

---
(i) Discarding An Event

\[ (\sigma, \varepsilon) \xrightarrow{u} (\sigma', \varepsilon') \]

if

- an \( E \)-event (instance of signal \( E \)) is ready in \( \varepsilon \) for object \( u \) of a class \( C \), i.e. if

\[ u \in \text{dom}(\sigma) \cap \mathcal{P}(C) \land \exists u_E \in \mathcal{P}(E) : u_E \in \text{ready}(\varepsilon, u) \]

- \( u \) is stable and in state machine state \( s \), i.e. \( \sigma(u)(\text{stable}) = 1 \) and \( \sigma(u)(st) = s \),

- but there is no corresponding transition enabled (all transitions incident with current state of \( u \) either have other triggers or the guard is not satisfied)

\[ \forall (s, F, \text{expr}, act, s') \in \rightarrow(SM_C) : F \neq E \lor I[\text{expr}](\sigma, u) = 0 \]

and

- in the system configuration, stability may change, \( u_E \) goes away, i.e.

\[ \sigma' = \sigma[u, \text{stable} \mapsto b] \setminus \{u_E \mapsto \sigma(u_E)\} \]

where \( b = 0 \) if and only if there is a transition with trigger ‘-’ enabled for \( u \) in \( (\sigma', \varepsilon') \).

- the event \( u_E \) is removed from the ether, i.e.

\[ \varepsilon' = \varepsilon \oplus u_E, \]

- consumption of \( u_E \) is observed, i.e.

\[ \text{cons} = \{u_E\}, \quad \text{Snd} = \emptyset. \]
Example: Discard

\[ G[x > 0]/x := y \]

\[ H/z := y/x \]

\( SM_C: \)

\[
\begin{array}{c}
\text{Init}\: \sigma: \\
\begin{array}{l}
x = 1, z = 0, y = 2 \\
\text{stable} = 1
\end{array}
\end{array}
\]

\( \varepsilon: \)

\[ (\text{signal}, \text{env}) \]

\( (\text{signal}) \)

\( G, J \)

\( n \)

\[
\begin{array}{c}
\text{C} \\
\begin{array}{l}
x, z : \text{Int} \\
y : \text{Int} \langle \text{env} \rangle
\end{array}
\end{array}
\]

\( (\text{signal}, \text{env}) \)

\( H \)

\( \langle \text{signal} \rangle \)

\( G, J \)

\( C \)

(ii) Dispatch

\[ (\sigma, \varepsilon) \xrightarrow{(\text{cons}, \text{Snd})} (\sigma', \varepsilon') \]

if

- \( u \in \text{dom}(\sigma) \cap \mathcal{D}(C) \)
- \( u \in \mathcal{D}(E), u_E \in \text{ready}(\varepsilon, u) \)
- \( \forall F, expr, act, s' \in (SM_C): F \neq E \lor I[expr](\sigma, u) = 0 \)
- \( \sigma'(u)(\text{stable}) = 1, \sigma(u)(st) = s, \sigma' = \sigma[u.st \mapsto s', u.stable \mapsto b] \}
- \( \varepsilon' = \varepsilon \ominus u_E \)
- \( \text{cons} = \{ u_E \}, \text{Snd} = \emptyset \)

and

- \( (\sigma', \varepsilon') \) results from applying \( t_{\text{act}} \) to \( (\sigma, \varepsilon) \) and removing \( u_E \) from the ether, i.e.

\[ (\sigma'', \varepsilon') \in t_{\text{act}}[u](\sigma', \varepsilon \ominus u_E), \]

\[ \sigma' = (\sigma''[u.st \mapsto s', u.stable \mapsto b, u.params_E \mapsto \emptyset])|_{\mathcal{D}(E)} \]

where \( b \) depends (see (i))

- Consumption of \( u_E \) and the side effects of the action are observed, i.e.

\[ \text{cons} = \{ u_E \}, \text{Snd} = \text{Observe}[u](\sigma, \varepsilon \ominus u_E). \]
Example: Dispatch

\[ SMC: \]

\[ G(x > 0) / x := y \]

\[ H / z := y / x \]

\[ \sigma: \]

\[ x = 1, z = 0, y = 2 \]

\[ st = s_1 \]

\[ stable = 1 \]

\[ \epsilon: \]

\[ G \text{ for } c \]

\[ (x_0, b) \]

\[ x^c \]

\[ x = 2, z = 0, y = 2 \]

\[ st = s_1 \]

\[ stable = 0 \]

\[ \sigma': \]

\[ \sigma' = \sigma / u.\text{params} \to \emptyset \]

\[ \sigma'' = \sigma' / u.\text{params} \to \emptyset \]

\[ cons = \{ u \} \]

\[ Snd = \text{Obs}_{u.\text{params}}(\sigma, \epsilon) \]

(iii) Continue Run-to-Completion

\[ (\sigma, \epsilon) \xrightarrow{(\text{cons}, \text{Snd})} (\sigma', \epsilon') \]

if

- there is an unstable object \( u \) of a class \( C \), i.e.

\[ u \in \text{dom}(\sigma) \cap \mathcal{P}(C) \]

\[ u \in \mathcal{P}(E) \]

- there is a transition without trigger enabled from the current state \( s = \sigma(u)(st) \), i.e.

\[ \exists (s, F, \text{expr}, \text{act}, s') \in (SMC): F = E \land I[\text{expr}](\sigma, u) = 1 \]

and

- \( (\sigma', \epsilon') \) results from applying \( t_{\text{act}} \) to \( (\sigma, \epsilon) \), i.e.

\[ (\sigma'', \epsilon') \in t_{\text{act}}[\sigma, \epsilon] \]

\[ \sigma' = \sigma'' / u.(st) \to s', u.\text{stable} \to b \]

where \( b \) depends as before.

- Only the side effects of the action are observed, i.e.

\[ cons = \emptyset, \quad Snd = \text{Obs}_{u.\text{params}}(\sigma, \epsilon). \]
**Example:** Commence (adapted) \[ x > 0 \implies x := x - 1; n! J \]

**SMC:**

\[
\begin{align*}
&G[x > 0]/x := y \\
&H/z := y/x
\end{align*}
\]

**H/z := y/x**

\[
\begin{align*}
&x = 2, z = 0, y = 2 \\
&st = s_2 \\
&stable = 0
\end{align*}
\]

\[
\begin{align*}
&\sigma' = \sigma \cup \{u_E \mapsto \{v_i \mapsto d_i \mid 1 \leq i \leq n\}, \quad \epsilon' = \epsilon \oplus (u, u_E)
\end{align*}
\]

\[
\begin{align*}
&\sigma' = \sigma' \mapsto u, st \mapsto s', \, stable \mapsto b \\
&\epsilon' = \epsilon \oplus (u, u_E)
\end{align*}
\]

\[
\begin{align*}
&\forall v \in V \forall u \in \text{dom}(\sigma) : \sigma'(u)(v) \neq \sigma(u)(v) \implies v \in V_{env}
\end{align*}
\]

\[
\begin{align*}
&\epsilon' = \epsilon
\end{align*}
\]

(iv) **Environment Interaction**

Assume that a set \( \mathcal{E}_{env} \subseteq \mathcal{E} \) is designated as environment events and a set of attributes \( V_{env} \subseteq V \) is designated as input attributes.

Then

\[
\begin{align*}
&\left(\sigma, \epsilon\right) \xrightarrow{\text{cons, Snd}} \left(\sigma', \epsilon'\right)
\end{align*}
\]

**if either (!)**

- an environment event \( E \in \mathcal{E}_{env} \) is spontaneously sent to an alive object \( u \in \text{dom}(\sigma) \), i.e.

\[
\begin{align*}
&\sigma' = \sigma \cup \{u_E \mapsto \{v_i \mapsto d_i \mid 1 \leq i \leq n\}, \quad \epsilon' = \epsilon \oplus (u, u_E)
\end{align*}
\]

where \( u_E \notin \text{dom}(\sigma) \) and \( \text{atr}(E) = \{v_1, \ldots, v_n\} \).

- Sending of the event is observed, i.e. \( \text{cons} = \emptyset, \, \text{Snd} = \{u_E,\} \).

**or**

- Values of input attributes change freely in alive objects, i.e.

\[
\begin{align*}
&\forall v \in V \forall u \in \text{dom}(\sigma) : \sigma'(u)(v) \neq \sigma(u)(v) \implies v \in V_{env}
\end{align*}
\]

and no objects appear or disappear, i.e. \( \text{dom}(\sigma') = \text{dom}(\sigma) \).

- \( \epsilon' = \epsilon \).
(v) Error Conditions

if, in (i), (ii), or (iii),

- \( I[expr] \) is not defined for \( \sigma \) and \( u \), or
- \( t_{act}[u] \) is not defined for \( (\sigma, \varepsilon) \),

and

- \( cons = \emptyset \), and \( Snd = \emptyset \).

Examples:

- \( E[x/0]/act \) to \( 82 \)
- \( E[true]/act \) to \( 83 \)
- \( E[expr]/x := x/0 \) to \( 82 \)
Example: Error Condition

\[ [x > 0]/x := x - 1; n! J \]

\[ \langle \langle \text{signal}, \text{env} \rangle \rangle H \]

\[ \langle \langle \text{signal} \rangle \rangle G, J \]

\[ C \]

\[ x, z: \text{Int} \quad y: \text{Int} \quad \langle \langle \text{env} \rangle \rangle \]

\[ \sigma: \]

\[ x = 0, z = 0, y = 27 \]

\[ \text{st} = s_2 \]

\[ \text{stable} = 1 \]

\[ \varepsilon: \]

\[ H \text{ for } e \]

- \( I[\text{expr}] \) not defined for \( \sigma \) and \( u \), or
- \( I[\text{act}] \) is not defined for \((\sigma, \varepsilon)\)
- \( \text{cons} = \emptyset \)
- \( \text{Snd} = \emptyset \)

Example Revisited

\[ C \]

\[ D \]

\[ \text{Nr.} \quad 1_C: C \quad 5_D: D \quad \varepsilon \quad \text{rule} \]

<table>
<thead>
<tr>
<th>Nr.</th>
<th>( x )</th>
<th>( n )</th>
<th>( \text{st} )</th>
<th>( \text{stable} )</th>
<th>( p )</th>
<th>( \text{st} )</th>
<th>( \text{stable} )</th>
<th>( \varepsilon )</th>
<th>rule</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>27</td>
<td>5_D</td>
<td>( s_1 )</td>
<td>1</td>
<td>1_C</td>
<td>( s_1 )</td>
<td>1</td>
<td>( 3_F, 1_C ), ( 2_E, 1_C )</td>
<td>( C )</td>
</tr>
<tr>
<td>1</td>
<td>( s_1 )</td>
<td>( s_2 )</td>
<td>1</td>
<td>( s_1 )</td>
<td>1</td>
<td>( 2_E, 1_C )</td>
<td>( C )</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>( s_2 )</td>
<td>( s_3 )</td>
<td>0</td>
<td>( s_1 )</td>
<td>1</td>
<td>( 2_F, s_2 )</td>
<td>( C )</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>( s_3 )</td>
<td>( s_4 )</td>
<td>1</td>
<td>( s_1 )</td>
<td>1</td>
<td>( 3_F, s_3 )</td>
<td>( G_F )</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>( s_3 )</td>
<td>( s_5 )</td>
<td>0</td>
<td>( s_1 )</td>
<td>0</td>
<td>-</td>
<td>( G_F )</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>( s_2 )</td>
<td>( s_6 )</td>
<td>0</td>
<td>( s_1 )</td>
<td>0</td>
<td>-</td>
<td>( G_F )</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
