Softwaretechnik / Software-Engineering

Lecture 09: Live Sequence Charts

2015-06-11

Prof. Dr. Andreas Podelski, Dr. Bernd Westphal

Albert-Ludwigs-Universität Freiburg, Germany
Contents & Goals

Last Lecture:

- Scenarios and Anti-Scenarios
- User Stories, Use Cases, Use Case Diagrams
- LSC: abstract and concrete syntax

This Lecture:

- **Educational Objectives:** Capabilities for following tasks/questions.
  - Which are the cuts and firedsets of this LSC?
  - Construct the TBA of a given LSC body.
  - Given a set of LSCs, which scenario/anti-scenario/requirement is formalised by them?
  - Formalise this positive scenario/anti-scenario/requirement using LSCs.

- **Content:**
  - Excursion: automata accepting infinite words
  - Cuts and Firedsets, automaton construction
  - existential LSCs, pre-charts, universal LSCs
  - Requirements Engineering: conclusions
Recall: LSC Body Syntax

LSC Body Example

- $L : l_{1,0} \prec l_{1,1} \prec l_{1,2} \prec l_{1,3}, \ l_{1,2} \prec l_{1,4}, \ l_{2,0} \prec l_{2,1} \prec l_{2,2} \prec l_{2,3}, \ l_{3,0} \prec l_{3,1} \prec l_{3,2}$
- $l_{1,1} \prec l_{2,1}, \ l_{2,2} \prec l_{1,2}, \ l_{2,3} \prec l_{1,3}, \ l_{3,2} \prec l_{1,4}, \ l_{2,2} \sim l_{3,1}$

- $I = \{\{l_{1,0}, l_{1,1}, l_{1,2}, l_{1,3}, l_{1,4}\}, \{l_{2,0}, l_{2,1}, l_{2,2}, l_{2,3}\}, \{l_{3,0}, l_{3,1}, l_{3,2}\}\}$

- $Msg = \{(l_{1,1}, A, l_{2,1}), (l_{2,2}, B, l_{1,2}), (l_{2,2}, C, l_{3,1}), (l_{2,3}, D, l_{1,3}), (l_{3,2}, E, l_{1,4})\}$

- $Cond = \{\{(l_{2,2}), c_2 \land c_3\}\}$

- $LocInv = \{(l_{1,1}, c_1, l_{1,2}, \bullet)\}$
LSC Semantics
• **Recall**: decision tables

• By the standard semantics, a decision table $T$ is software, $\llbracket T \rrbracket = \{ \sigma_0 \xrightarrow{\alpha_1} \sigma_1 \xrightarrow{\alpha_2} \sigma_2 \cdots \mid \cdots \}$ is a set of computation paths.

• **Recall**: Decision tables as software specification:

![Decision Table Diagram](image)

- We want **the same** for LSCs.

- We will give a **procedure** to construct for each LSC $\mathcal{L}$ an **automaton** $B(\mathcal{L})$. The language (or semantics) of $\mathcal{L}$ is the set of comp. paths **accepted** by $B(\mathcal{L})$. Thus an LSC is also software.

- **Problem**: computation paths may be infinite $\rightarrow$ Büchi acceptance.
Excursion: Symbolic Büchi Automata
From Finite Automata to Symbolic Büchi Automata

\[ \mathcal{A}: \Sigma = \{0, 1\} \]

\[ \mathcal{B}: \Sigma = \{0, 1\} \]

\[ \text{Büchi infinite words} \]

\[ \mathcal{A}_{\text{sym}}: \Sigma = (\{x\} \to \mathbb{N}) \]

\[ \mathcal{B}_{\text{sym}}: \Sigma = (\{x\} \to \mathbb{N}) \]

\[ \text{Büchi infinite words} \]

\[ \text{symbolic even}(x) \]

\[ \text{symbolic odd}(x) \]

\[ w = 0101 \in \text{Lang}(\mathcal{A}) \]

\[ w = 0101 \in \text{Lang}(\mathcal{A}) \]

\[ \text{symbolic even}(x) \]

\[ \text{symbolic odd}(x) \]

\[ w = 01001000\ldots \in \text{Lang}(\mathcal{B}) \]

\[ w = 01001000\ldots \in \text{Lang}(\mathcal{B}) \]

\[ \text{Lang}(\mathcal{A}_{\text{sym}}) = \{ x_0, x_1, x_2, x_3, \ldots \mid \text{even}(x_0) \land \forall i \geq 1 \text{ even}(x_i) \land \text{odd}(x_{i+1}) \} \]
Definition. A **Symbolic Büchi Automaton** (TBA) is a tuple

\[ \mathcal{B} = (\mathcal{C}, Q, q_{ini}, \rightarrow, Q_F) \]

where

- \( \mathcal{C} \) is a set of atomic propositions,
- \( Q \) is a finite set of **states**,
- \( q_{ini} \in Q \) is the initial state,
- \( \rightarrow \subseteq Q \times \Phi(\mathcal{C}) \times Q \) is the finite **transition relation**. Each transitions \((q, \psi, q') \in \rightarrow\) from state \(q\) to state \(q'\) is labelled with a formula \(\psi \in \Phi(\mathcal{C})\).
- \( Q_F \subseteq Q \) is the set of **fair** (or accepting) states.
Run of TBA

**Definition.** Let $\mathcal{B} = (\mathcal{C}, Q, q_{ini}, \to, Q_F)$ be a TBA and

$$w = \sigma_1, \sigma_2, \sigma_3, \ldots \in (\mathcal{C} \to \mathbb{B})^\omega$$

an infinite word, each letter is a valuation of $\mathcal{C}_\mathcal{B}$.

An infinite sequence

$$\varrho = q_0, q_1, q_2, \ldots \in Q^\omega$$

of states is called **run** of $\mathcal{B}$ over $w$ if and only if

- $q_0 = q_{ini}$,
- for each $i \in \mathbb{N}_0$ there is a transition $(q_i, \psi_i, q_{i+1}) \in \to$ s.t. $\sigma_i \models \psi_i$.

**Example:**

$$\omega = (x=0)(x=1)(x=4)(x=5)(x=8)(x=9)\ldots$$

$\mathcal{B}_{sym}$: $\Sigma = (\{x\} \to \mathbb{N})$

**Example:**

$L = q_1, 7_2, 9_1, 9_2, 7_1, 7_2, \ldots$
**The Language of a TBA**

**Definition.**
We say TBA $B = (C, Q, q_{ini}, \rightarrow, Q_F)$ accepts the word $w = (\sigma_i)_{i \in \mathbb{N}_0} \in (C \rightarrow B)^\omega$ if and only if $B$ has a run

$$\varrho = (q_i)_{i \in \mathbb{N}_0}$$

over $w$ such that fair (or accepting) states are **visited infinitely often** by $\varrho$, i.e., such that

$$\forall i \in \mathbb{N}_0 \exists j > i : q_j \in Q_F.$$ 

We call the set $\text{Lang}(B) \subseteq (C \rightarrow B)^\omega$ of words that are accepted by $B$ the **language of** $B$. 
Example

run: \( \rho = q_0, q_1, q_2, \ldots \in Q^\omega \) s.t. \( \sigma_i \models \psi_i, i \in \mathbb{N}_0 \).
LSC Semantics: TBA Construction
**Definition.** Let \( ((\mathcal{L}, \preceq, \sim), \mathcal{I}, \text{Msg}, \text{Cond}, \text{LocInv}, \Theta) \) be an LSC body.

A non-empty set \( \emptyset \neq C \subseteq \mathcal{L} \) is called a **cut** of the LSC body iff \( C \)

- is **downward closed**, i.e. 
  \[ \forall l, l' \in \mathcal{L} \cdot l' \in C \land l \preceq l' \implies l \in C, \]

- is **closed** under **simultaneity**, i.e.
  \[ \forall l, l' \in \mathcal{L} \cdot l' \in C \land l \sim l' \implies l \in C, \]

- comprises at least **one location per instance line**, i.e.
  \[ \forall I \in \mathcal{I} \cdot C \cap I \neq \emptyset. \]

The temperature function is extended to cuts as follows:

\[
\Theta(C) = \begin{cases} 
\text{hot} & \text{if } \exists l \in C \cdot (\nexists l' \in C \cdot l \prec l') \land \Theta(l) = \text{hot} \\
\text{cold} & \text{otherwise}
\end{cases}
\]

that is, \( C \) is **hot** if and only if at least one of its maximal elements is hot.
Cut Examples

\[ \emptyset \neq C \subseteq \mathcal{L} \text{ — downward closed — simultaneity closed — at least one loc. per instance line} \]
$\emptyset \neq C \subseteq \mathcal{L} \quad \text{— downward closed — simultaneity closed — at least one loc. per instance line}$
A Successor Relation on Cuts

The partial order “≤” and the simultaneity relation “∼” of locations induce a **direct successor relation** on cuts of \( \mathcal{L} \) as follows:

**Definition.**
Let \( C \subseteq \mathcal{L} \) bet a cut of LSC body \(((\mathcal{L}, \preceq, \sim), \mathcal{I}, \text{Msg}, \text{Cond}, \text{LocInv}, \Theta)\). A set \( \emptyset \neq \mathcal{F} \subseteq \mathcal{L} \) is called **fired-set** \( \mathcal{F} \) of \( C \) if and only if

- \( C \cap \mathcal{F} = \emptyset \) and \( C \cup \mathcal{F} \) is a cut, i.e. \( \mathcal{F} \) is closed under simultaneity,
- all locations in \( \mathcal{F} \) are **direct ≺-successors** of the front of \( C \), i.e.
  \[
  \forall l \in \mathcal{F} \ \exists l' \in C \ \bullet \ l' \prec l \ \land \ (\nexists l'' \in C \ \bullet \ l' \prec l''),
  \]
- locations in \( \mathcal{F} \), that lie on the same instance line, are **pairwise unordered**, i.e.
  \[
  \forall l \neq l' \in \mathcal{F} \ \bullet \ (\exists I \in \mathcal{I} \ \bullet \ \{l, l'\} \subseteq I) \implies l \not\preceq l' \land l' \not\preceq l,
  \]
- for each asynchronous message reception in \( \mathcal{F} \), the corresponding sending is already in \( C \),
  \[
  \forall (l, E, l') \in \text{Msg} \ \bullet \ l' \in \mathcal{F} \implies l \in C.
  \]

The cut \( C' = C \cup \mathcal{F} \) is called **direct successor of \( C \ via \( \mathcal{F} \)**, denoted by \( C \sim_\mathcal{F} C' \).
$C \cap F = \emptyset$ — $C \cup F$ is a cut — only direct $\prec$-successors — same instance line on front pairwise unordered — sending of asynchronous reception already in
$C \cap \mathcal{F} = \emptyset$ — $C \cup \mathcal{F}$ is a cut — only direct $\prec$-successors — same instance line on front pairwise unordered — sending of asynchronous reception already in
The TBA $\mathcal{B}(\mathcal{L})$ of LSC $\mathcal{L}$ over $C$ and $\mathcal{E}$ is $(C, Q, q_{ini}, \rightarrow, Q_F)$ with

- $Q$ is the set of cuts of $\mathcal{L}$, $q_{ini}$ is the instance heads cut,
- $C = C \cup \mathcal{E}_{!\,?}$, where $\mathcal{E}_{!\,?} = \{E!, E? \mid E \in \mathcal{E}\}$,
- $\rightarrow$ consists of loops, progress transitions (from $\rightsquigarrow \mathcal{F}$), and legal exits (cold cond./local inv.),
- $Q_F = \{C \in Q \mid \Theta(C) = \text{cold} \lor C = \mathcal{L}\}$ is the set of cold cuts and the maximal cut.
Recall: The TBA $\mathcal{B}(\mathcal{L})$ of LSC $\mathcal{L}$ is $(C, Q, q_{ini}, \rightarrow, Q_F)$ with

- $Q$ is the set of cuts of $\mathcal{L}$, $q_{ini}$ is the instance heads cut,
- $C = C \cup \{ E! , E? \mid E \in \mathcal{E} \}$,
- $\rightarrow$ consists of loops, progress transitions (from $\rightsquigarrow F$), and legal exits (cold cond./local inv.),
- $F = \{ C \in Q \mid \Theta(C) = \text{cold} \lor C = \mathcal{L} \}$ is the set of cold cuts.

So in the following, we “only” need to construct the transitions’ labels:

$$\rightarrow = \{ (q, \psi_{\text{loop}}(q), q) \mid q \in Q \} \cup \{ (q, \psi_{\text{prog}}(q, q'), q') \mid q \rightsquigarrow F q' \} \cup \{ (q, \psi_{\text{exit}}(q), \mathcal{L}) \mid q \in Q \}$$
So in the following, we “only” need to construct the transitions’ labels:

\[
\rightarrow = \{(q, \psi_{\text{loop}}(q), q) \mid q \in Q\} \cup \{(q, \psi_{\text{prog}}(q, q'), q') \mid q \leadsto q'\} \cup \{(q, \psi_{\text{exit}}(q), L) \mid q \in Q\}
\]

\[
\psi_{\text{loop}}(q) = \psi_{\text{Msg}}(q) \land \psi_{\text{LocInv, hot}}(q) \land \psi_{\text{LocInv, cold}}(q)
\]

\[
\psi_{\text{exit}}(q) = \\
(\psi_{\text{loop}}^{\text{hot}}(q) \land \neg \psi_{\text{LocInv, cold}}(q)) \\
\lor \bigvee_{1 \leq i \leq n} \psi_{\text{prog}}^{\text{hot}}(q, q_i) \land \\
(\neg \psi_{\text{LocInv, cold, hot}}(q, q_i) \lor \neg \psi_{\text{cond, cold}}(q, q_i))
\]

\[
\psi_{\text{prog}}(q, q_n) = \\
\psi_{\text{prog}}^{\text{hot}}(q, q_n) \land \psi_{\text{cond, cold}}^{\text{hot}}(q, q_n) \land \psi_{\text{cond, cold}}^{\text{hot}}(q, q_n) \\
\land \psi_{\text{LocInv, cond, cold}}^{\text{hot}}(q, q_n) \land \psi_{\text{LocInv, cond, cold}}^{\text{hot}}(q, q_n)
\]

\[
\psi_{\text{prog}}(q, q_n) = \psi_{\text{Msg}}^{\text{hot}}(q, q_n)
\]
Loop Condition

\[ \psi_{loop}(q) = \psi^{\text{Msg}}(q) \land \psi^{\text{LocInv}}_{\text{hot}}(q) \land \psi^{\text{LocInv}}_{\text{cold}}(q) \]

- \[ \psi^{\text{Msg}}(q) = \neg \bigvee_{1 \leq i \leq n} \psi^{\text{Msg}}(q, q_i) \land (\text{strict} \implies \land_{\psi \in \mathcal{E} \cap \text{Msg}(\mathcal{L})} \neg \psi)\]

- \[ \psi^{\text{LocInv}}_{\theta}(q) = \land_{\ell = (l, \nu, \phi, l', \nu', l') \in \text{LocInv}, \Theta(\ell) = \theta, \ell \text{ active at } q} \phi \]

A location \( l \) is called front location of cut \( C \) if and only if \( \not\exists l' \in \mathcal{L} \bullet l < l' \).

Local invariant \((l_0, \nu_0, \phi, l_1, \nu_1)\) is active at cut (!) \( q \) if and only if \( l_0 \leq l \leq l_1 \) for some front location \( l \) of cut (!) \( q \).

- \( \text{Msg}(\mathcal{F}) = \{E! \mid (l, E, l') \in \text{Msg}, l \in \mathcal{F}\} \cup \{E? \mid (l, E, l') \in \text{Msg}, l' \in \mathcal{F}\} \)

- \( \text{Msg}(\mathcal{F}_1, \ldots, \mathcal{F}_n) = \bigcup_{1 \leq i \leq n} \text{Msg}(\mathcal{F}_i) \)
Progress Condition

\[ \psi_{\text{hot}}(q, q_i) = \psi_{\text{Msg}}(q, q_n) \land \psi_{\text{Cond}}(q, q_n) \land \psi_{\text{LocInv}, \bullet}(q_n) \]

1. The msgs. of fired set \( F \), not msg. combination of any other fired set of \( Z \)

\[ \psi_{\text{Msg}}(q, q_i) = \bigwedge_{\psi \in \text{Msg}(q_i \setminus q)} \psi \land \bigwedge_{j \neq i} \bigwedge_{\psi \in (\text{Msg}(q_j \setminus q) \setminus \text{Msg}(q_i \setminus q))} \neg \psi \]

2. \( \psi_{\text{Cond}}(q, q_i) = \bigwedge_{\gamma=(L,\phi) \in \text{Cond}, \Theta(\gamma)=\theta, L \cap (q_i \setminus q) \neq \emptyset} \phi \)

3. \( \psi_{\text{LocInv}, \bullet}(q, q_i) = \bigwedge_{\lambda=(l,\iota,\phi,l',\iota') \in \text{LocInv}, \Theta(\lambda)=\theta, \lambda \bullet\text{-active at } q_i} \phi \)

Local invariant \((l_0, \iota_0, \phi, l_1, \iota_1)\) is \( \bullet\text{-active} \) at \( q \) if and only if

- \( l_0 \prec l \prec l_1 \), or
- \( l = l_0 \land \iota_0 = \bullet \), or
- \( l = l_1 \land \iota_1 = \bullet \)

for some front location \( l \) of cut (!) \( q \).
Example
Finally: The LSC Semantics

A **full LSC** \( \mathcal{L} = (((\mathcal{L}, \preceq, \sim), I, \text{Msg}, \text{Cond}, \text{LocInv}, \Theta), ac_0, am, \Theta_{\mathcal{L}}) \) consist of

- **body** \( ((\mathcal{L}, \preceq, \sim), I, \text{Msg}, \text{Cond}, \text{LocInv}, \Theta) \),

- **activation condition** \( ac_0 \in \Phi(C) \), **strictness flag** \( strict \) (otherwise called **permissive**),

- **activation mode** \( am \in \{\text{initial, invariant}\} \),

- **chart mode** **existential** \( (\Theta_{\mathcal{L}} = \text{cold}) \) or **universal** \( (\Theta_{\mathcal{L}} = \text{hot}) \).

Concrete syntax:
Finally: The LSC Semantics

A full LSC $\mathcal{L} = (((\mathcal{L}, \preceq, \sim), \mathcal{I}, \text{Msg}, \text{Cond}, \text{LocInv}, \Theta), ac_0, am, \Theta_\mathcal{L})$ consist of

- **body** $((\mathcal{L}, \preceq, \sim), \mathcal{I}, \text{Msg}, \text{Cond}, \text{LocInv}, \Theta)$,
- **activation condition** $ac_0 \in \Phi(C)$, **strictness flag** strict (otherwise called permissive)
- **activation mode** $am \in \{\text{initial, invariant}\}$,
- **chart mode** existential ($\Theta_\mathcal{L} = \text{cold}$) or universal ($\Theta_\mathcal{L} = \text{hot}$).

A set of words $W \subseteq (C \rightarrow B)^\omega$ is **accepted** by $\mathcal{L}$ if and only if

<table>
<thead>
<tr>
<th>$\Theta_\mathcal{L}$</th>
<th>$am = \text{initial}$</th>
<th>$am = \text{invariant}$</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>cold</strong></td>
<td>$\exists w \in W \bullet w^0 \models ac \land$</td>
<td>$\exists w \in W \exists k \in \mathbb{N}_0 \bullet w^k \models ac \land$</td>
</tr>
<tr>
<td></td>
<td>$w^0 \models \psi_{\text{Cond}}^{\text{hot}}(\emptyset, C_0) \land w/1 \in \text{Lang}(B(\mathcal{L}))$</td>
<td>$w^k \models \psi_{\text{Cond}}^{\text{hot}}(\emptyset, C_0) \land w/k+1 \in \text{Lang}(B(\mathcal{L}))$</td>
</tr>
<tr>
<td><strong>hot</strong></td>
<td>$\forall w \in W \bullet w^0 \models ac \implies$</td>
<td>$\forall w \in W \forall k \in \mathbb{N}_0 \bullet w^k \models ac \implies$</td>
</tr>
<tr>
<td></td>
<td>$w^0 \models \psi_{\text{Cond}}^{\text{hot}}(\emptyset, C_0) \land w/1 \in \text{Lang}(B(\mathcal{L}))$</td>
<td>$w^k \models \psi_{\text{Cond}}^{\text{hot}}(\emptyset, C_0) \land w/k+1 \in \text{Lang}(B(\mathcal{L}))$</td>
</tr>
</tbody>
</table>

where $ac = ac_0 \land \psi_{\text{Cond}}^{\text{cold}}(\emptyset, C_0) \land \psi_{\text{Msg}}(\emptyset, C_0)$; $C_0$ is the minimal (or instance heads) cut.
References

