Software Design, Modelling and Analysis in UML

Lecture 19: Live Sequence Charts III

Prof. Dr. Andreas Podelski, Dr. Bernd Westphal

Albert-Ludwigs-Universität Freiburg, Germany

Content

• Live Sequence Charts
  • Semantics
    • TBA Construction for LSC Body
      • Cuts, Firedsets
      • Signal / Attribute Expressions
      • Loop / Progress Conditions
    • Excursion: Büchi Automata
  • Language of a Model
  • Full LSCs
    • Existential and Universal
    • Pre-Charts
    • Forbidden Scenarios
  • LSCs and Tests
Plan:

(i) Given an LSC $\mathcal{L}$ with body

$((L, \preceq, \sim), I, \text{Msg}, \text{Cond}, \text{LocInv}, \Theta)$,

(ii) construct a TBA $B_{\mathcal{L}}$, and

(iii) define language $\mathcal{L}(\mathcal{L})$ of $\mathcal{L}$ in terms of $\mathcal{L}(B_{\mathcal{L}})$,

in particular taking activation condition and activation mode into account.

(iv) define language $\mathcal{L}(M)$ of a UML model.

- Then $M \models \mathcal{L}$ (universal) if and only if $\mathcal{L}(M) \subseteq \mathcal{L}(\mathcal{L})$.

And $M \models \mathcal{L}$ (existential) if and only if $\mathcal{L}(M) \cap \mathcal{L}(\mathcal{L}) \neq \emptyset$. 
Formal LSC Semantics: It’s in the Cuts!

Definition.

Let $(\mathcal{L}, \preceq, \sim), I, \text{Msg}, \text{Cond}, \text{LocInv}, \Theta)$ be an LSC body.

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

- it is **downward closed**, i.e. $\forall l, l' \in C \land l \preceq l' \implies l \in C$,
- it is **closed under simultaneity**, i.e. $\forall l, l' \in C \land l \sim l' \implies l \in C$, and
- it comprises at least **one location per instance line**, i.e. $\forall i \in I \exists l \in C \cdot il = i$.

The **temperature function** is extended to cuts as follows:

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

that is, $C$ is **hot** if and only if at least one of its maximal elements is hot.
$\emptyset \neq C \subseteq L$ -- downward closed -- simultaneity closed -- at least one loc. per instance line
A Successor Relation on Cuts

The partial order \( \preceq \) and the simultaneity relation \( \sim \) of locations induce a direct successor relation on cuts of an LSC body as follows:

**Definition.** Let \( C \subseteq L \) be a cut of LSC body \( ((L, \preceq, \sim), \mathcal{I}, \text{Msg}, \text{Cond}, \text{LocInv}, \Theta) \).

A set \( \emptyset \neq F \subseteq L \) of locations is called **fired-set** \( F \) of cut \( C \) if and only if:

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

The cut \( C' = C \cup F \) is called **direct successor of \( C \)** via \( F \), denoted by \( C \leadsto_F C' \).

Successor Cut Example

\( C \cap F = \emptyset - C \cup F \) is a cut – only direct \( \prec \)-successors – same instance line on front pairwise unordered – sending of asynchronous reception already in \( F \).
**Successor Cut Example**

\[ C \cap F = \emptyset - C \cup F \text{ is a cut} - \text{only direct} - \text{successors} - \text{same instance line} \text{ on front pairwise unordered} - \text{sending of asynchronous reception already in} \]

**Language of LSC Body: Example**

The TBA \( B_L \) of LSC \( L \) over \( \mathbb{R} \) is \( (\text{Expr}_L(X), X, Q, q_{init}, \rightarrow, Q_F) \) with:

- \( Q \) is the set of cuts of \( L \), \( q_{init} \) is the instance heads cut.
- \( \text{Expr}_L(X) = \text{Expr}_{\mathbb{R}}(\mathcal{E}, X) \) (for considered signature \( \mathcal{S} \)).
- \( \rightarrow \) consists of loops, progress transitions (by \( \rightarrow \)), and legal exits (cold cond./local inv.).
- \( Q_F = \{ C \in Q \mid \Theta(C) = \text{cold} \lor C = L \} \) is the set of cold cuts and the maximal cut.
Signal and Attribute Expressions

• Let $\mathcal{S} = (\mathcal{T}, \mathcal{E}, V, \text{atr}, \mathcal{E})$ be a signature and $X$ a set of logical variables,

• The signal and attribute expressions $\text{Expr}_\mathcal{S}(\mathcal{E}, X)$ are defined by the grammar:

$$\psi ::= \text{true} \mid \psi \mid E \mid E \mathord{!} x,y \mid E \mathord{?} x,y \mid \neg \psi \mid \psi_1 \lor \psi_2$$

where $\text{expr} \in \text{Expr}_\mathcal{S}$, $E \in \mathcal{E}$, $x, y \in X$ (or keyword $\text{env}$).

• We use $E \mathord{!?} (X) := \{ E \mathord{!} x,y \mid E \in \mathcal{E}, x, y \in X \}$ to denote the set of event expressions over $\mathcal{E}$ and $X$.

**TBA Construction Principle**

Recall: The TBA $\mathcal{B}(L)$ of LSC $L$ is $(\text{Expr}_\mathcal{B}(X), X, Q, q_{ini}, \rightarrow, Q_F)$ with

• $Q$ is the set of cuts of $L$, $q_{ini}$ is the instance heads cut,

• $\text{Expr}_\mathcal{B} = \Phi \cup \mathcal{E}(X)$,

• $\rightarrow$ consists of loops, progress transitions (from $\rightarrow_F$), and legal exits (cold cond./local inv.),

• $F = \{ C \in Q \mid \Theta(C) = \text{cold} \lor C = L \}$ is the set of cold cuts.

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

$$\rightarrow = \{(q, \psi_{\text{loop}}(q), q) \mid q \in Q \} \cup \{(q, \psi_{\text{prog}}(q, q'), q') \mid q \rightarrow_F q' \} \cup \{(q, \psi_{\text{exit}}(q), L) \mid q \in Q \}$$
“Only” construct the transitions’ labels:
\[ \rightarrow = \{(q, \psi_{\text{loop}}(q), q) \mid q \in Q\} \cup \{(q, \psi_{\text{prog}}(q, q'), q') \mid q \leadsto q'\} \cup \{(q, \psi_{\text{exit}}(q), L) \mid q \in Q\} \]

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

\[ \psi_{\text{exit}}(q) = \left( \psi_{\text{msg}}(q) \land \psi_{\text{LocInv}}^\text{hot}(q) \land \psi_{\text{LocInv}}^\text{cold}(q) \right) \]

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

Loop Condition

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

- \[ \psi_{\text{msg}}(q) = \neg \bigvee_{1 \leq i \leq n} \psi_{\text{msg}}(q, q_i) \land \psi_{\text{LocInv}}^\text{hot}(q) \land \psi_{\text{LocInv}}^\text{cold}(q) \]

- \[ \psi_{\text{LocInv}}^\text{hot}(q) = \bigwedge_{(l, \theta, \phi, l', \iota, \phi') \in \text{LocInv}, \Theta(l) = \theta, \iota} \text{active at } q \]

A location \( l \) is called front location of cut \( C \) if and only if \( \exists l' \in L \land l \prec l' \).

Local invariant \( (l_0, \iota_0, \phi_0, l_1, \iota_1) \) is active at cut \( l \) if and only if \( l_0 \leq l < l_1 \) for some front location \( l \) of cut \( C \).

- \[ \text{Msg}(F) = \{E_{x_1, x_1'} \mid (l, E, l') \in \text{Msg}, l \in F\} \cup \{E_{x_1, x_1'} \mid (l, E, l') \in \text{Msg}, l' \in F\} \]

- \( x_1 \in X \) is the logical variable associated with the instance line \( I \) which includes \( l \), i.e. \( l \in I \).

- \[ \text{Msg}(F_1, \ldots, F_n) = \bigcup_{1 \leq i \leq n} \text{Msg}(F_i) \]
Progress Condition

\[ \psi_{\text{prog}}^{\text{hot}}(q, q_i) = \psi^{\text{Mag}}(q, q_n) \land \psi^{\text{Cond}}_{\text{hot}}(q, q_n) \land \psi^{\text{LocInv}}_{\text{hot}}(q, q_n) \]

- \[ \psi^{\text{Mag}}(q, q_i) = \bigwedge_{\psi \in \text{Mag}(q_i \setminus q)} \psi \land \bigwedge_{j \neq i} \psi \in \text{Mag}(q_j \setminus q) \land \text{Mag}(q_i \setminus q) \land \neg \psi \]
  \[ \land \left( \text{strict } \implies \bigwedge_{\psi \in \text{Mag}(L) \setminus \text{Mag}(F_i) \setminus q} \neg \psi \right) \]
  \[ =: \psi_{\text{new}}(q, q_i) \]

- \[ \psi^{\text{Cond}}_{\text{hot}}(q, q_i) = \bigwedge_{\gamma = (L, \phi) \in \text{Cond}, \Theta(\gamma) = \theta, L \setminus q_i \notin \phi} \psi \]

- \[ \psi^{\text{LocInv}}_{\text{hot}}(q, q_i) = \bigwedge_{\lambda = (l, \iota, \phi, l', \iota') \in \text{LocInv}, \Theta(\lambda) = \theta \land \lambda \text{-active at } q_i} \psi \]

Local invariant \((l_0, c_0, \phi, l_1, l_1)\) is •-active at \(q\) if and only if
- \(l_0 < l < l_1\), or
- \(l = l_0 \land c_0 = \bullet\), or
- \(l = l_1 \land c_1 = \bullet\)

for some front location \(l\) of cut (!) \(q\).

Example

Using logical variables \(x, y, z\) for the instances lines (from left to right).
Excursion: Büchi Automata
Symbolic Büchi Automata

Definition. A **Symbolic Büchi Automaton** (TBA) is a tuple

\[ B = (\text{Expr}_B(X), X, Q_{\text{ini}}, \rightarrow, Q_F) \]

where

- \( X \) is a set of logical variables.
- \( \text{Expr}_B(X) \) is a set of Boolean expressions over \( X \).
- \( Q \) is a finite set of states.
- \( Q_{\text{ini}} \in Q \) is the initial state.
- \( \rightarrow \subseteq Q \times \text{Expr}_B(X) \times Q \) is the transition relation. Transitions \((q, \psi, q') \) from \( q \) to \( q' \) are labelled with an expression \( \psi \in \text{Expr}_B(X) \).
- \( Q_F \subseteq Q \) is the set of **fair** (or accepting) states.
Definition. Let $X$ be a set of logical variables and let $\text{Expr}_B(X)$ be a set of Boolean expressions over $X$.

A set $(\Sigma, \models \cdot \cdot)$ is called an **alphabet** for $\text{Expr}_B(X)$ if and only if
\begin{itemize}
    \item for each $\sigma \in \Sigma$,
    \item for each expression $\text{expr} \in \text{Expr}_B$, and
    \item for each valuation $\beta : X \rightarrow \mathcal{P}(X)$ of logical variables,
\end{itemize}
\[ \text{either } \sigma \models _\beta \text{expr} \text{ or } \sigma \not\models _\beta \text{expr}. \]

(\text{$\sigma$ satisfies (or does not satisfy) $\text{expr}$ under valuation $\beta$})

An infinite sequence
\[ w = (\sigma_i)_{i \in \mathbb{N}_0} \in \Sigma^\omega \]
over $(\Sigma, \models \cdot \cdot)$ is called **word** (for $\text{Expr}_B(X)$).

---

**Run of TBA over Word**

Definition. Let $B = (\text{Expr}_B(X), X, Q, q_{ini}, \rightarrow, Q_F)$ be a TBA and
\[ w = \sigma_1, \sigma_2, \sigma_3, \ldots \]
a word for $\text{Expr}_B(X)$. An infinite sequence
\[ q = q_0, q_1, q_2, \ldots \in Q^\omega \]
is called **run of $B$ over $w$** under valuation $\beta : X \rightarrow \mathcal{P}(X)$ if and only if
\begin{itemize}
    \item $q_0 = q_{ini}$,
    \item for each $i \in \mathbb{N}_0$ there is a transition
    \[ (q_i, \psi_i, q_{i+1}) \in \rightarrow \]
    such that $\sigma_i \models _\beta \psi_i$.
\end{itemize}

**Example:**

\[ B_{sym}: \]
\[ \Sigma = (\{x\} \rightarrow \mathbb{N}) \]
\[ \text{even}(x), \text{odd}(x) \]
\[ q_1, q_2, q_3 \]

\[ \text{even}(x) \]
\[ \text{odd}(x) \]
\[ \Sigma = \{x\} \rightarrow \mathbb{N} \]
Definition.
We say TBA $B = (Expr_B(X), X, Q, q_{init}, \rightarrow, Q_F)$ accepts the word
$$w = (\sigma_i)_{i \in \mathbb{N}_0} \in (Expr_B \rightarrow B)^\omega$$
if and only if $B$ has a run
$$q = (q_i)_{i \in \mathbb{N}_0}$$
over $w$ such that fair (or accepting) states are visited infinitely often by $q$.
i.e., such that
$$\forall i \in \mathbb{N}_0 \exists j > i : q_j \in Q_F.$$ We call the set $L(B) \subseteq (Expr_B \rightarrow B)^\omega$ of words that are accepted by $B$ the language of $B$.

Example:

$B_{sym}$:

$\Sigma = ([x] \rightarrow \mathbb{N})$

References
References
