Software Design, Modelling and Analysis in UML

Lecture 17: Live Sequence Charts II

2012-01-31

Prof. Dr. Andreas Podelski, Dr. Bernd Westphal
Albert-Ludwigs-Universität Freiburg, Germany

Contents & Goals

Last Lecture:
- Reflective vs. constructive description of behaviour
- Live Sequence Charts: syntax, intuition

This Lecture:
- Educational Objectives: Capabilities for following tasks/questions.
  - What does this LSC mean?
  - Are this UML model’s state machines consistent with the interactions?
  - Please provide a UML model which is consistent with this LSC.
  - What is: activation, hot/cold condition, pre-chart, etc.?

- Content:
  - Symbolic Büchi Automata (TBA) and its (accepted) language.
  - LSC formal semantics.
Recall: Example

- Whenever the CrossingCtrl has consumed a 'secreq' event
- then it shall finally send 'lights on' and 'barrier down' to LightsCtrl and BarrierCtrl,
- if LightsCtrl is not 'operational' when receiving that event, the rest of this scenario doesn’t apply; maybe there’s another LSC for that case.
- if LightsCtrl is 'operational' when receiving that event, it shall reply with 'lights ok' within 1–3 time units,
- the BarrierCtrl shall reply with 'barrier ok' within 1–5 time units, during this time (dispatch time not included) it shall not be in state 'MvUp',
- 'lights ok' and 'barrier ok' may occur in any order.
- After having consumed both, CrossingCtrl may reply with 'done' to the environment.
Recall: LSC Body – Abstract Syntax

Let $\Theta = \{\text{hot, cold}\}$. An **LSC body** is a tuple 

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

where

- $I$ is a finite set of **instance lines**, each associated with a class $\mathcal{C}$.
- $(\mathcal{L}, \preceq)$ is a finite, non-empty, partially ordered set of **locations**, each $l \in \mathcal{L}$ is associated with a temperature $\theta(l) \in \Theta$ and an instance line $i_l \in I$.
- $\sim \subseteq \mathcal{L} \times \mathcal{L}$ is an **equivalence relation** on locations, the **simultaneity** relation.
- $\mathcal{S} = (\mathcal{P}, \mathcal{E}, V, \mathcal{A}, \mathcal{W})$ is a signature,
- $\text{Msg} \subseteq \mathcal{L} \times \mathcal{E} \times \mathcal{L}$ is a set of **asynchronous messages** with $(l, b, l') \in \text{Msg}$ only if $l \sim l'$,
- Not: instantaneous messages — could be linked to method/operation calls.
- $\text{Cond} \subseteq (\mathcal{P} \setminus \emptyset) \times \mathcal{E} \times \Theta$ is a set of **conditions** with $(l, \text{expr}, \theta) \in \text{Cond}$ only if $l \sim l'$ for all $l, l' \in \mathcal{L}$.
- $\text{LocInv} \subseteq \mathcal{L} \times \{\circ, \bullet\} \times \mathcal{E} \times \Theta \times \mathcal{L} \times \{\circ, \bullet\}$ is a set of **local invariants**.

**Example**

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

$\text{Msg} \subseteq \mathcal{L} \times \mathcal{E} \times \mathcal{L}$

$\text{Cond} \subseteq (\mathcal{P} \setminus \emptyset) \times \mathcal{E} \times \Theta$

$\text{LocInv} \subseteq \mathcal{L} \times \{\circ, \bullet\} \times \mathcal{E} \times \Theta \times \mathcal{L} \times \{\circ, \bullet\}$$

Example:

$I = \{i_1, i_2, i_3\}$, $\Theta = \{\text{hot}, \text{cold}\}$, $\mathcal{S} = \{C_1, C_2\}$

$\mathcal{L} = \{l_1, l_2, l_3\}$

$\text{LocInv} = \{l_1, l_2, l_3\}$

$\text{Cond} = \{C_1, C_2\}$

$\text{Msg} = \{C_1, C_2\}$

$\mathcal{S} = \{(l_1, \text{expr}, \text{hot}), ...\}$

$\mathcal{E} = \{(l_1, l_2, \text{hot}), ...\}$
Recall: Well-Formedness

Bondedness/no floating conditions: (could be relaxed a little if we wanted to)

- For each location \( l \in \mathcal{L} \), if \( l \) is the location of
  - a condition, i.e.
    \[ \exists (L, expr, \theta) \in \text{Cond} : l \in L, \]
  - a local invariant, i.e.
    \[ \exists \left(l_1, i_1, expr, \theta, l_2, i_2\right) \in \text{LocInv} : l \in \{l_1, l_2\}, \]
  then there is a location \( l' \) equivalent to \( l \) which is the location of
  - a message, i.e.
    \[ \exists (l_1, b, l_2) \in \text{Msg} : l \in \{l_1, l_2\}, \]
  - an instance head, i.e. \( l' \) is minimal wrt. \( \preceq \).

Note: if messages in a chart are cyclic, then there doesn’t exist a partial order (so such charts don’t even have an abstract syntax).

Course Map
Live Sequence Charts Semantics

TBA-based Semantics of LSCs

Plan:
- Given an LSC $L$ with body
  $$(I, (\mathcal{L}, \preceq), \sim, \mathcal{S}, \text{Msg}, \text{Cond}, \text{LocInv})$$
- Construct a TBA $B_L$ — taking the cuts of $L$ as states.
- Define $\mathcal{L}(L)$ in terms of $\mathcal{L}(B_L)$,
  in particular taking activation condition and activation mode into account.
Formal LSC Semantics: It’s in the Cuts

- Let \( (I, (\mathcal{L}, \preceq), \sim, \mathcal{S}, \text{Msg}, \text{Cond}, \text{LocInv}) \) be an LSC body.
- A non-empty set \( \emptyset \neq C \subseteq \mathcal{L} \)
  is called a cut of the LSC body if and only if
  - it is downward closed, i.e. \( \forall l, l' : l' \in C \land l \preceq l' \implies l \in C \),
  - it is closed under simultaneity, i.e. \( \forall l, l' : l' \in C \land l \sim l' \implies l \in C \), and
  - it comprises at least one location per instance line, i.e. \( \forall i \in I \ \exists l \in C : i_l = i \).

A cut \( C \) is called hot, denoted by \( \theta(C) = \text{hot} \), if and only if at least one of its maximal elements is hot, i.e. if
\[
\exists l \in C : \theta(l) = \text{hot} \land \nexists l' \in C : l < l'
\]
Otherwise, \( C \) is called cold, denoted by \( \theta(C) = \text{cold} \).
Examples: Cut or Not Cut? Hot/Cold?

(i) non-empty set $\emptyset \neq C \subseteq \mathcal{L}$.
(ii) downward closed, i.e.
$$\forall l,l' : l' \in C \land l \preceq l' \Rightarrow l \in C$$
(iii) closed under simultaneity, i.e.
$$\forall l,l' : l' \in C \land l \sim l' \Rightarrow l \in C$$
(iv) at least one location per instance line, i.e.
$$\forall i \in I \exists l \in C : i_l = i$$

\[ C_0 = \emptyset \]
\[ C_1 = \{l_{1,0}, l_{2,0}, l_{3,0}\} \]
\[ C_2 = \{l_{1,1}, l_{2,1}, l_{3,0}\} \]
\[ C_3 = \{l_{1,0}, l_{1,1}\} \]
\[ C_4 = \{l_{1,0}, l_{1,1}, l_{2,0}, l_{3,0}\} \]
\[ C_5 = \{l_{1,0}, l_{1,1}, l_{2,0}, l_{2,1}, l_{3,0}\} \]
\[ C_6 = \mathcal{L} \setminus \{l_{1,3}, l_{2,3}\} \]
\[ C_7 = \mathcal{L} \setminus /C_4 \]

A Successor Relation on Cuts

The partial order of $(\mathcal{L}, \preceq)$ and the simultaneity relation "~" induce a **direct successor relation** on cuts of $\mathcal{L}$ as follows:

- Let $C, C' \subseteq \mathcal{L}$ be cuts. $C'$ is called direct successor of $C$
  - via fired-set $F$, denoted by $C \leadsto_F C'$, if and only if
    - $F \neq \emptyset$,
    - $C' \setminus C = F$,
    - for each message reception in $F$, the corresponding sending is already in $C$,
    - locations in $F$, that lie on the same instance line, are pairwise unordered, i.e.
      $$\forall l, l' \in F : l \neq l' \land i_l = i_{l'} \Rightarrow l \not\preceq l' \land l' \not\preceq l$$
  - $F$ is immediately closed under simultaneity, (\sim)
  - In other words: locations in $F$ are direct $\preceq$-successors of locations in $C$, i.e.
    $$\forall l' \in F \exists l \in C : l \prec l' \land \exists l'' \in C : l' \prec l'' \prec l$$
Successor Cut Examples

(i) \( F \neq \emptyset \),
(ii) \( C' \setminus C = F \),
(iii) message send before receive,
(iv) locations on same instance line unordered, i.e.
\[ \forall l, l' \in F : l \neq l' \land \mu_l = \mu_{l'} \implies l \not\preceq l' \land l' \not\preceq l \]

\[ \begin{align*}
C_1 & \coloneqq (\sigma_1, \text{cons}_1, \text{Snd}_1) \\
C_2 & \coloneqq (\sigma_2, \text{cons}_2, \text{Snd}_2) \\
C_3 & \coloneqq (\sigma_3, \text{cons}_3, \text{Snd}_3)
\end{align*} \]

\[ v = 0 \]

\begin{align*}
l_{1,0} & \preceq A \preceq l_{1,1} \\
l_{2,0} & \preceq B \preceq l_{2,1} \\
l_{2,2} & \preceq C \preceq l_{2,3} \\
l_{3,0} & \preceq D \preceq l_{3,1} \\
l_{3,2} & \preceq E \preceq l_{3,3}
\end{align*}

\[ v = 0 \]

\[ \begin{align*}
A & \coloneqq [x] \preceq l_{2,3} \\
B & \coloneqq [x] \preceq l_{1,1} \\
C & \coloneqq [x] \preceq l_{2,2} \\
D & \coloneqq [x] \preceq l_{3,1} \\
E & \coloneqq [x] \preceq l_{1,0}
\end{align*} \]

Idea: Accepting Words by Advancing the Cut

Let \( w = (\sigma_i, \text{cons}_i, \text{Snd}_i)_{i \in \mathbb{N}_0} \) be a word over \( \mathcal{S} \) and \( \mathcal{D} \).

Intuitively (and for now disregarding cold conditions), an LSC body \((I, (\mathcal{S}, \preceq), \sim, \mathcal{D}, \text{Msg}, \text{Cond}, \text{LocInv})\) is supposed to accept \( w \) (under valuation \( \beta \)) if and only if there exists a sequence

\[ C_0 \leadsto F_1 C_1 \leadsto F_2 C_2 \cdots \leadsto F_n C_n \]

and indices \( i_1 < \cdots < i_n \) such that

- \( C_0 \) consists of the instance heads,
- for all \( 1 \leq j < n \),
  - for all \( i_j \leq k < i_{j+1} \), \((\sigma_k, \text{cons}_k, \text{Snd}_k)\) satisfies (under \( \beta \)) the hold condition of \( C_{j-1} \),
  - \((\sigma_{i_j}, \text{cons}_{i_j}, \text{Snd}_{i_j})\) satisfies (under \( \beta \)) the transition condition of \( F_j \),
- \( C_n \) is cold,
- for all \( i_n < k \), \((\beta_k, \mu_k, t_{i_n})\) satisfies (under \( \beta \)) the hold condition of \( C_n \).
Symbolic Büchi Automata

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

\[ B = (\mathit{Expr}_B, X, Q, q_{\text{init}}, \rightarrow, Q_F) \]

where

- \( \mathit{Expr}_B \) is a set of expressions over logical variables from \( X \),
- \( Q \) is a finite set of \textit{states}, \( q_{\text{init}} \) the initial state,
- \( \rightarrow \subseteq Q \times \mathit{Expr}_B \times Q \) is the \textit{transition relation}.
- Transitions \((q, \mathit{expr}, q')\) from \( q \) to \( q' \) are labelled with a constraint \( \mathit{expr} \in \mathit{Expr}_B \) over the signals and \( X \) variables.
- \( Q_F \subseteq Q \) is the set of \textit{fair} (or accepting) states.
**TBA Example**

\[(\text{Expr}_B, X, Q, q_{\text{ini}}, \rightarrow, Q_F)\]

\[\begin{align*}
q_1 & \quad \neg a(x, y) \\
q_2 & \quad b(y) \\
q_3 & \quad \neg b(y) \\
q_4 & \quad d(y, x) \\
q_5 & \quad true
\end{align*}\]

\[\begin{align*}
\text{Expr}_B = \{ a(x, y), \neg a(x, y), \neg a(y, y), \neg a(x, y), \neg a(x, y), \neg a(y, y) \} \\
Q = \{ q_1, q_2, q_3, q_4, q_5 \} \\
q_{\text{ini}} = q_1 \\
Q_f = \{ q_5 \}
\end{align*}\]

---

**Word**

**Definition.** Let \(\text{Expr}_B\) be a set of expressions over logical variables \(X\), and let \(\Sigma\) be the set of interpretation functions of \(\text{Expr}_B\), i.e.

\[
\Sigma = \text{Expr}_B \times (X \rightarrow \mathcal{P}(X)) \rightarrow \{0, 1\}.
\]

For \(\sigma \in \Sigma\), we write \(\sigma \models_\beta \text{expr}\) if and only if \(\sigma(\text{expr}, \beta) = 1\).

A **word** over \(\text{Expr}_B\) is an infinite sequence of interpretations of \(\text{Expr}_B\)

\[
(\sigma_i)_{i \in \mathbb{N}_0} \in \Sigma^\omega.
\]

\[\omega: \quad \sigma_0 \models_B a(x, y), \quad \beta = \{ x \mapsto 1, y \mapsto 2 \}
\]

\[\sigma_1 \models_C \neg a, \quad \sigma_2 \models e(x) \quad \text{(nothing \(e(x)\))}\]
**Run of TBA over Word**

**Definition.** Let $B = (\text{Expr}_B, X, Q, q_\text{ini}, \rightarrow, Q_\text{F})$ be a TBA and $w = (\sigma_i)_{i \in \mathbb{N}_0} \in \Sigma^\omega$ a word over $\text{Expr}_B$.

An infinite sequence

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

is called **run** of $B$ over $w$ under valuation $\beta : X \rightarrow \mathcal{P}(X)$ if and only if

- $q_0 = q_\text{ini}$,
- for each $i \in \mathbb{N}_0$ there is a transition $(q_i, \psi_i, q_{i+1}) \in \rightarrow$ such that $\sigma_i \models \beta \psi_i$.

**Run or Not Run Examples**

$\rho = (q_i)_{i \in \mathbb{N}_0}$, $q_0 = q_\text{ini}$, $\forall i \in \mathbb{N}_0 \exists (q_i, \psi_i, q_{i+1}) \in \rightarrow : (\sigma_i, \text{cons}_i, \text{Snd}_i) \models \beta \psi_i$

$\omega$:

- $\sigma_0 \models \neg a(x,y)$
- $\sigma_0 \models \neg \sigma_0(y)$
- $\sigma_0 \models \neg a(y)$
- $\sigma_0 \models \neg b(y)$
- $\sigma_0 \models \neg \sigma_0(x)$
- $\sigma_0 \models \neg d(y,x)$

$\rho = q_0 q_1 q_2 \ldots$
The Language of a TBA

Definition.
We say $B = (Expr_B, X, Q, q_{ini}, \rightarrow, Q_F)$ accepts $w$ (under valuation $\beta : X \rightarrow \mathcal{P}(X)$) if and only if $B$ has a run $(q_i)_{i \in \mathbb{N}_0}$ over $w$ such that fair (or accepting) states are visited infinitely often, that is, $\forall i \in \mathbb{N}_0 \exists j > i : q_j \in Q_F$.

We call the set $L_\beta(B)$ of words over $\mathcal{P}$ that are accepted by $B$ under $\beta$ the language of $B$.

Language of the Example TBA

$L_\beta(B)$ consists of the words

$$(\sigma_i, \text{Snd}_i, \text{cons}_i)_{i \in \mathbb{N}_0}$$

where there exist $0 \leq n < m < k < \ell$ such that

- for $0 \leq i < n$, $\sigma_i \not\models_B a(x,y)$
- $\sigma_n \models_B a(x,y)$
- for $n < i < m$, $\sigma_i \not\models_B b(y)$
- $\sigma_m \models_B b(y) \land c$ and
  - for $m < i < k$, $\sigma_i \not\models_B d(y,x)$
  - $\sigma_k \models_B d(y,x)$
  - for $k < i < \ell$, $\sigma_i \not\models_B e(x)$
  - $\sigma_\ell \models_B e(x)$, or
- $\sigma_m \models_B b(y) \land \neg c$
Recall Idea: Accepting Words by Advancing the Cut

Let \( w = (\sigma_i, \text{cons}_i, \text{Snd}_i)_{i \in \mathbb{N}_0} \) be a word over \( \mathcal{S} \) and \( \mathcal{D} \).

Intuitively (and for now disregarding cold conditions), an LSC body \( (I, (\mathcal{S}, \preceq), \sim, \mathcal{S}, \text{Msg}, \text{Cond}, \text{LocInv}) \) is supposed to accept \( w \) (under valuation \( \beta \)) if and only if there exists a sequence

\[
C_0 \sim_F C_1 \sim_F C_2 \ldots \sim_F C_n
\]

and indices \( i_1 < \cdots < i_n \) such that

- \( C_0 \) consists of the instance heads,
- for all \( 1 \leq j < n \),
  - for all \( i_j \leq k < i_{j+1} \), \( (\sigma_k, \text{cons}_k, \text{Snd}_k) \) satisfies (under \( \beta \)) the hold condition of \( C_{j-1} \),
  - \( (\sigma_{i_j}, \text{cons}_{i_j}, \text{Snd}_{i_j}) \) satisfies (under \( \beta \)) the transition condition of \( F_j \),
- \( C_n \) is cold,
- for all \( i_n < k \), \( (\beta_k, \mu_k, t_k) \) satisfies (under \( \beta \)) the hold condition of \( C_n \).
Language of LSC Body

The language of the body

\[(I, (\mathcal{L}, \preceq), \sim, \mathcal{F}, \text{Msg, Cond, LocInv})\]

of LSC \(L\) is the language of the TBA

\[\mathcal{B}_L = (\text{Expr}_B, X, Q, q_{\text{ini}}, \rightarrow, Q_F)\]

with

- \(\text{Expr}_B = \text{Expr}_\mathcal{F}(V, \mathcal{E}(\mathcal{P}))\)
- \(Q\) is the set of cuts of \((\mathcal{L}, \preceq)\), \(q_{\text{ini}}\) is the instance heads cut,
- \(Q_F = \{C \in Q \mid \theta(C) = \text{cold}\}\) is the set of cold cuts of \((\mathcal{L}, \preceq)\),
- \(\rightarrow\) as defined in the following, consisting of
  - loops \((q, \psi, q)\),
  - progress transitions \((q, \psi, q')\), and
  - legal exits \((q, \psi, \mathcal{L})\).

Language of LSC Body: Intuition

\[\mathcal{B}_L = (\text{Expr}_B, X, Q, q_{\text{ini}}, \rightarrow, Q_F)\] with

- \(\text{Expr}_B = \text{Expr}_\mathcal{F}(V, \mathcal{E}(\mathcal{P}))\)
- \(Q\) is the set of cuts of \((\mathcal{L}, \preceq)\), \(q_{\text{ini}}\) is the instance heads cut,
- \(F = \{C \in Q \mid \theta(C) = \text{cold}\}\) is the set of cold cuts,
- \(\rightarrow\) consists of
  - loops \((q, \psi, q)\),
  - progress transitions \((q, \psi, q')\), and
  - legal exits \((q, \psi, \mathcal{L})\).
Signal and Integer Expressions

Let $\mathcal{S} = (\mathcal{P}, \mathcal{C}, V, atr)$ be a signature and $X$ a set of logical variables. The signal and integer expressions $Expr_{\mathcal{S}}(V, \mathcal{E}(\mathcal{S}))$ over $\mathcal{S}$ are defined by the grammar:

$$\psi ::= true \mid expr \mid E! x, y \mid E? x \mid \neg \psi \mid \psi_1 \lor \psi_2,$$

where $expr \in Expr_{\mathcal{S}}, E \in \mathcal{E}, x, y \in X$.

Satisfaction of Signal and Integer Expressions

Let $(\sigma, cons, Snd) \in (\Sigma^\mathcal{S} \times 2^{\mathcal{P}(\mathcal{S})} \times Evs(\mathcal{E}, \mathcal{P}) \times 2^{\mathcal{P}(\mathcal{S})} \times Evs(\mathcal{E}, \mathcal{P}) \times Evs(\mathcal{E}, \mathcal{P}))$ be a letter of a word over $\mathcal{S}$ and $\mathcal{D}$ and let $\beta : X \rightarrow \mathcal{D}(\mathcal{C})$ be a valuation of the logical variables in $X$.

- $(\sigma, cons, Snd) \models_\beta true$
- $(\sigma, cons, Snd) \models_\beta \neg \psi$ if and only if not $(\sigma, cons, Snd) \models_\beta \psi$
- $(\sigma, cons, Snd) \models_\beta \psi_1 \lor \psi_2$ if and only if
  - $(\sigma, cons, Snd) \models_\beta \psi_1$ or $(\sigma, cons, Snd) \models_\beta \psi_2$
- $(\sigma, cons, Snd) \models_\beta expr$ if and only if $[expr](\sigma, \beta) = 1$
- $(\sigma, cons, Snd) \models_\beta E! x, y$ if and only if $(\beta(x), (E, \vec{d}), \beta(y)) \in Snd$
- $(\sigma, cons, Snd) \models_\beta E? x$ if and only if $(\beta(x), (E, \vec{d})) \in cons$
Satisfaction of Signal and Integer Expressions

Let \((\sigma, \text{cons}, \text{Snd}) \in (\Sigma^\delta \times 2^{\Sigma(C)} \times \text{Evs}(\delta, \mathcal{D}) \times 2^{\Sigma(C)} \times \text{Evs}(\delta, \mathcal{D}) \times 2^{\Sigma(C)})\) be a letter of a word over \(\mathcal{S}\) and \(\mathcal{D}\) and let \(\beta : \mathcal{X} \to 2\) be a valuation of the logical variables in \(\mathcal{X}\).

- \((\sigma, \text{cons}, \text{Snd}) \models_\beta \text{true}\)
- \((\sigma, \text{cons}, \text{Snd}) \models_\beta \neg \psi\) if and only if not \((\sigma, \text{cons}, \text{Snd}) \models_\beta \psi\)
- \((\sigma, \text{cons}, \text{Snd}) \models_\beta \psi_1 \lor \psi_2\) if and only if \((\sigma, \text{cons}, \text{Snd}) \models_\beta \psi_1\) or \((\sigma, \text{cons}, \text{Snd}) \models_\beta \psi_2\)
- \((\sigma, \text{cons}, \text{Snd}) \models_\beta \text{expr}\) if and only if \(I[\text{expr}]((\sigma, \beta) = 1\)
- \((\sigma, \text{cons}, \text{Snd}) \models_\beta E_{x,y}\) if and only if \((\beta(x), (E, \vec{d}), \beta(y)) \in \text{Snd}\)
- \((\sigma, \text{cons}, \text{Snd}) \models_\beta E_{x,y}^2\) if and only if \((\beta(x), (E, \vec{d})) \in \text{cons}\)

Observation: if the semantics has “forgotten” the sender at consumption time, then we have to disregard it here (straightforwardly fixed if desired). Other view: we could choose to disregard the sender.

Example: TBA over Signal and Integer Expressions
Some Helper Functions

• **Messages of a location:**

\[
B(l) := \{ b \in B \mid \exists l', b, l \in \text{Msg} \land (l', b, l) \in \text{Msg} \},
\]

\[
B(\{l_1, \ldots, l_n\}) := B(l_1) \cup \cdots \cup B(l_n).
\]

• **Constraints** relevant at cut \( q \):

\[
\psi(q) = \{ \psi \mid \exists l \in q, l' \notin q \mid (l, \psi, \theta, l') \in \text{LocInv} \lor (l', \psi, \theta, l) \in \text{LocInv} \},
\]

\[
B(\mathcal{L}) = \{ (c, e), (e, c) \}\]

Some More Helper Functions

• **Constraints** relevant when moving from \( q \) to cut \( q' \):

\[
\psi(q, q') = \{ \psi \mid \exists l \in q' \setminus q, l' \in \mathcal{L}, \theta \in \Theta \mid \]

\[
(l, \bullet, expr, \theta, l') \in \text{LocInv} \lor (l', expr, \theta, l, \bullet) \in \text{LocInv}
\]

\[
\cup \{ \psi \mid \exists l \in q, l' \notin q', \theta \in \Theta \mid \]

\[
(l, expr, \theta, l') \in \text{LocInv} \lor (l', expr, \theta, l) \in \text{LocInv}
\]

\[
\cup \{ \psi \mid \exists L \subseteq \mathcal{L}, \theta \in \Theta \mid (L, \psi, \theta) \in \text{Cond} \land L \cap (q' \setminus q) \neq \emptyset \}.
\]
**Even More Helper Functions**

- **Cold constraints** relevant when moving from \( q \) to cut \( q' \):

\[
\psi_{\text{cold}}(q, q') = \{ \psi \mid \exists l \in q' \setminus q, l' \in \mathcal{L} \mid (l, \bullet, \text{expr}, \text{cold}, l') \in \text{LocInv} \lor (l', \text{expr}, \text{cold}, l, \bullet) \in \text{LocInv} \} \\
\cup \{ \psi \mid \exists l \in q, l' \notin q' \mid (l, \text{expr}, \text{cold}, l') \in \text{LocInv} \lor (l', \text{expr}, \text{cold}, l, \bullet) \in \text{LocInv} \} \\
\cup \{ \psi \mid \exists L \subseteq \mathcal{L} \mid (L, \psi, \text{cold}) \in \text{Cond} \land L \cap (q' \setminus q) \neq \emptyset \}
\]

---

**Recall: Intuition**

\( B_L = (\text{Expr}_B, X, Q, q_{ini}, \rightarrow, Q_F) \) with
- \( \text{Expr}_B = \text{Expr}_S(V, \beta(\mathcal{L})) \)
- \( Q \) is the set of cuts of \( (\mathcal{L}, \subseteq), \) \( q_{ini} \) is the instance heads cut,
- \( F = \{ C \in Q \mid \theta(C) = \text{cold} \} \) is the set of cold cuts,
- \( \rightarrow \) consists of
  - loops \( (q, \psi, q) \),
  - progress transitions \( (q, \psi, q') \), and
  - legal exits \( (q, \psi, \mathcal{L}) \).
Loops

- How long may we legally stay at a cut \( q \)?
- **Intuition**: those \( (\sigma_i, \text{cons}_i, \text{Snd}_i) \) are allowed to fire the self-loop \( (q, \psi, q) \) where
  - \( \text{cons}_i \cup \text{Snd}_i \) comprises only irrelevant messages:
    - **weak mode**: (parameter)
      no message from a direct successor cut is in,
    - **strict mode**:
      no message occurring in the LSC is in,
  - \( \sigma_i \) satisfies the local invariants active at \( q \)
  - And nothing else.
- **Formally**: Let \( F := F_1 \cup \cdots \cup F_n \) be the union of the firedsets of \( q \).

\[
\psi := \bigwedge_i \{ B(F_i) \wedge \psi(q_i) \}. \\
\text{weak mode}
\]

\[
\psi := \bigwedge_i \{ B(F_i) \wedge \neg \bigvee_j \{ B(F_j) \wedge \psi(q_j) \} \}. \\
\text{strict mode: add } \neg \left( \bigvee_j \{ \psi(q_j) \} \right)
\]

(no message from LSC is allowed)

Progress

- When do we move from \( q \) to \( q' \)?
- **Intuition**: those \( (\sigma_i, \text{cons}_i, \text{Snd}_i) \) fire the progress transition \( (q, \psi, q') \) for which there exists a firedset \( F \) such that \( q \leadsto_F q' \) and
  - \( \text{cons}_i \cup \text{Snd}_i \) comprises exactly the messages that distinguish \( F \) from other firedsets of \( q \) (weak mode), and in addition no message occurring in the LSC is in \( \text{cons}_i \cup \text{Snd}_i \) (strict mode),
  - \( \sigma_i \) satisfies the local invariants and conditions relevant at \( q' \).
- **Formally**: Let \( F, F_1, \ldots, F_n \) be the firedset of \( q \) and \( q \leadsto_F q' \) (unique).

\[
\psi := \bigwedge_i \{ B(F_i) \wedge \neg \bigvee_j \{ B(F_j) \cup \cdots \cup B(F_n) \} \wedge B(F_i) \wedge \psi(q_i, q') \}. \\
\text{weak mode}
\]

\[
\psi := \bigwedge_i \{ B(F_i) \wedge \text{the messages in firedset to} \} \\
\text{and no other firedset}
\]

\[
\text{respect conditions and \( \sigma_i \) invariants relevant at } q'.
\]
Legal Exits

- When do we take a legal exit from $q$?

  **Intuition:** those $(\sigma_i, \text{cons}_i, \text{Snd}_i)$ fire the legal exit transition $(q, \psi, \mathcal{L})$ for which there exists a firedset $F$ and some $q'$ such that $q \rightarrow_F q'$ and

  - $\text{cons}_i \cup \text{Snd}_i$ comprises exactly the messages that distinguish $F$ from other firedsets of $q$ (weak mode), and in addition no message occurring in the LSC is in $\text{cons}_i \cup \text{Snd}_i$ (strict mode).

  **Formally:** Let $F_1, \ldots, F_n$ be the firedset of $q$ with $q \rightarrow_{F_i} q_i$.

  - $\psi := \bigwedge_{i=1}^n B(F_i) \land \neg \left( \bigvee_{i=1}^n \left( B(F_1) \cup \cdots \cup B(F_n) \right) \setminus B(F_i) \right) \land \bigvee \psi_{\text{cold}}(q,q_i)$.

Example
Finally: The LSC Semantics

A full LSC $L$ consist of

- a body $(I, (L, \leq), \sim, J, \text{Msg}, \text{Cond}, \text{LocInv})$,
- an activation condition (here: event) $ac \in B$,
- an activation mode, either initial or invariant,
- a chart mode, either existential (cold) or universal (hot).

A set $W$ of timed words over $B$ and $V$ satisfies $L$, denoted $W \models L$, iff $L$ satisfies

- universal (= hot), initial, and
  \[ \forall w \in W \ \forall \beta : X \rightarrow \text{dom}(w_0) \bullet w \text{ activates } L \implies w \in L(B_L). \]
- universal (= hot), invariant, and
  \[ \forall w \in W \ \forall k \in \mathbb{N}_0 \ \forall \beta : X \rightarrow \text{dom}(w_k) \bullet w/k \text{ activates } L \implies w/k \in L(B_L). \]
- existential (= cold), initial, and
  \[ \exists w \in W \ \exists \beta : X \rightarrow \text{dom}(w_0) \bullet w \text{ activates } L \wedge w \in L(B_L). \]
- existential (= cold), invariant, and
  \[ \exists w \in W \ \exists k \in \mathbb{N}_0 \ \exists \beta : X \rightarrow \text{dom}(w_k) \bullet w/k \text{ activates } L \wedge w/k \in L(B_L). \]
Interactions as Reflective Description

- In UML, reflective (temporal) descriptions are subsumed by interactions.
- A UML model \( M = (\mathcal{D}, \mathcal{M}, \mathcal{OD}, \mathcal{I}) \) has a set of interactions \( \mathcal{I} \).
- An interaction \( I \in \mathcal{I} \) can be (OMG claim: equivalently) diagrammed as
  - sequence diagram,
  - timing diagram, or
  - communication diagram (formerly known as collaboration diagram).
Interactions as Reflective Description

- In UML, reflective (temporal) descriptions are subsumed by interactions.
- A UML model $\mathcal{M} = (\mathcal{C}D, \mathcal{I}M, \mathcal{O}D, \mathcal{I})$ has a set of interactions $\mathcal{I}$.
- An interaction $\mathcal{I} \in \mathcal{I}$ can be (OMG claim: equivalently) diagrammed as
  - sequence diagram, timing diagram, or
  - communication diagram (formerly known as collaboration diagram).
**Why Sequence Diagrams?**

**Most Prominent:** Sequence Diagrams — with long history:
- **Message Sequence Charts**, standardized by the ITU in different versions, often accused to lack a formal semantics.
- **Sequence Diagrams** of UML 1.x

Most severe drawbacks of these formalisms:
- **unclear interpretation:** example scenario or invariant?
- **unclear activation:** what triggers the requirement?
- **unclear progress requirement:** must all messages be observed?
- **conditions** merely comments
- **no means to express forbidden scenarios**

**Thus: Live Sequence Charts**

- **SDs of UML 2.x** address some issues, yet the standard exhibits unclarities and even contradictions [Harel and Maoz, 2007, Störrle, 2003]
- For the lecture, we consider Live Sequence Charts (LSCs) [Damm and Harel, 2001, Klose, 2003, Harel and Marelly, 2003], who have a common fragment with UML 2.x SDs [Harel and Maoz, 2007]
- **Modelling guideline:** stick to that fragment.
Side Note: Protocol Statemachines

Same direction: call orders on operations
- “for each C instance, method f() shall only be called after g() but before h()”

Can be formalised with protocol state machines.

References
References


