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: Live Sequence Charts Syntax
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

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

where

- $I$ is a finite set of **instance lines**, each associated with a class $C \in \mathcal{C}$
- $(\mathcal{L}, \preceq)$ 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,
- $\mathcal{I} = (\mathcal{I}, \mathcal{C}, V, \text{atr}, \mathcal{U})$ 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.
- $\text{Cond} \subseteq (2^{\mathcal{L} \setminus \emptyset}) \times \text{Expr} \times \Theta$ is a set of **conditions** with $(L, \text{expr}, \theta) \in \text{Cond}$ only if $l \sim l'$ for all $l, l' \in L$,
- $\text{LocInv} \subseteq \mathcal{L} \times \{\circ, \bullet\} \times \text{Expr} \times \Theta \times \mathcal{L} \times \{\circ, \bullet\}$ is a set of **local invariants**,
Example

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

\[ \text{Msg} \subseteq \mathcal{L} \times \mathcal{E} \times \mathcal{L} \]

\[ \text{Cond} \subseteq (2^{\mathcal{L}} \setminus \emptyset) \times \text{Expr} \times \Theta \]

\[ \text{LocInv} \subseteq \mathcal{L} \times \{\circ, \bullet\} \times \text{Expr} \times \Theta \times \mathcal{L} \times \{\circ, \bullet\} \]

\[ I = \{x, y, z\}, \quad C(x) = C_1, \ldots \]

\[ \mathcal{L} = \{ \text{hot}, \text{cold} \}, \ldots \}

\[ \leq \subseteq \mathcal{L} \times \mathcal{L}: \{ \text{hot} \leq \text{hot}, \ldots \}

\[ \leq_0 \subseteq \mathcal{L} \times \mathcal{L}: \{ \text{hot} \leq \text{hot}, \ldots \}

\[ \text{Msg} = \{ (l_2, A_1, l_{23}), \ldots \}

\[ \sim = \{ (l_4, l_{23}) \}

\[ \text{Cond} = \{ (l_4, l_{23}, \{x > 5\}, \text{hot}) \}, \ldots \}

\[ \text{LocInv} = \{ (l_{01}, 0, (v = 0), \text{cold}, l_{02}, \circ) \} \]
**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.
    \[ \exists (l_1, i_1, expr, \theta, l_2, i_2) \in \text{LocInv} : l \in \{l_1, l_2\}, \]

  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).
Live Sequence Charts Semantics
Plan:

- Given an LSC $L$ with body

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

- Construct a TBA $\mathcal{B}_L$ — taking the cuts of $L$ as states.

- Define $\mathcal{L}(L)$ in terms of $\mathcal{L}(\mathcal{B}_L)$, in particular taking activation condition and activation mode into account.
Formal LSC Semantics: It’s in the Cuts

- Let \((I, (\mathcal{L}, \preceq), \sim, \mathcal{L}, \text{Msg}, \text{Cond}, \text{LocInv})\) 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. \( \forall i \in I \exists l \in C : i_l = i \).
Formal LSC Semantics: It’s in the Cuts

- Let \((I, (\mathcal{L}, \preceq), \sim, \mathcal{L}, \text{Msg}, \text{Cond}, \text{LocInv})\) 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
  \[
  \exists l \in C : \theta(l) = \text{hot} \land \nexists l' \in C : l \prec l'
  \]
  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.
\[
\forall l, l' : l' \in C \land l \preceq l' \implies l \in C
\]

(iii) **closed under simultaneity**, i.e.
\[
\forall l, l' : l' \in C \land l \sim l' \implies l \in C
\]

(iv) **at least one location per instance line**, i.e.
\[
\forall i \in I \ \exists l \in C : i_l = i,
\]

- \( C_0 = \emptyset \)
- \( C_1 = \{l_{1,0}, l_{2,0}, l_{3,0}\} \)
- \( C_2 = \{l_{1,1}, l_{2,1}, l_{3,0}\} \)
- \( C_3 = \{l_{1,0}, l_{1,1}\} \)
- \( C_4 = \{l_{1,0}, l_{1,1}, l_{2,0}, l_{3,0}\} \)
- \( C_5 = \{l_{1,0}, l_{1,1}, l_{2,0}, l_{2,1}, l_{3,0}\} \)
- \( C_6 = \mathcal{L} \setminus \{l_{1,3}, l_{2,3}\} \)
- \( C_7 = \mathcal{L} \)
A Successor Relation on Cuts

The partial order of $\langle \mathcal{L}, \leq \rangle$ and the simultaneity relation $\sim$ 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 $C \sim_F C'$, 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.
    $$\forall l, l' \in F : l \neq l' \land i_l = i_{l'} \implies l \not\leq l' \land l' \not\leq l$$

- **Note:** $F$ is immediately closed under simultaneity. ($\sim$)

- In other words: locations in $F$ are direct $\leq$-successors of locations in $C$, i.e.
  $$\forall l' \in F \exists l \in C : l < l' \land \exists l'' \in C : l' < l'' < l$$
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. 
\[ \forall l, l' \in F : l \neq l' \land i_l = i_{l'} \implies l \not\preceq l' \land l' \not\preceq l \]
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 \( \mathcal{L} \) and \( \mathcal{D} \).

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

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

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, \( C_n = \mathcal{L} \)
- for all \( i_n < k \), \( (\beta_k, \mu_{i_j}, t_{i_j}) \) satisfies (under \( \beta \)) the **hold condition** of \( C_n \).
Excursus: Symbolic Büchi Automata (over Signature)
Definition. A **Symbolic Büchi Automaton** (TBA) is a tuple \[ B = (\mathcal{Expr}_B, X, Q, q_{ini}, \rightarrow, Q_F) \]
where
- \( \mathcal{Expr}_B \) is a set of expressions over logical variables from \( X \),
- \( Q \) is a finite set of **states**, \( q_{ini} \) the initial state,
- \( \rightarrow \subseteq Q \times \mathcal{Expr}_B \times Q \) is the **transition relation**. Transitions \( (q, expr, q') \) from \( q \) to \( q' \) are labelled with a constraint \( expr \in \mathcal{Expr}_B \) over the **signals** and **variables**.
- \( Q_F \subseteq Q \) is the set of **fair** (or accepting) states.
TBA Example

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

$\begin{align*}
\text{Expr}_3 & = \{ a(x,y), \\
& \quad \neg a(x,y), \\
& \quad \neg b(y), \\
& \quad b(y) \land c, \\
& \quad \neg d(y, x), \\
& \quad d(y, x), \\
& \quad \neg e(x), \\
& \quad e(x), \\
& \quad b(y) \land \neg c \} \\
\text{Q} & = \{ q_1, \ldots, q_5 \} \\
q_{ini} & = q_1 \\
Q_F & = \{ q_5 \} 
\end{align*}$
**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 \( \text{Expr}_B \)

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

\[
\omega:\quad \sigma_0 \models_\beta a(x,y) , \quad \beta = \emptyset \times \{1\} , \ y \mapsto 27
\]

\[
\sigma_1 \models_\beta c , \quad \sigma_1 \models e(x) \quad \text{(nothing else) }
\]
Definition. Let $B = (Expr_B, X, Q, q_{ini}, \rightarrow, Q_F)$ be a TBA and

$$w = (\sigma_i)_{i \in \mathbb{N}_0} \in \Sigma^\omega$$

a word over $Expr_B$.

An infinite sequence

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

is called run of $B$ over $w$ under valuation $\beta : X \rightarrow D(X)$ if and only if

- $q_0 = q_{ini}$,
- for each $i \in \mathbb{N}_0$ there is a transition $(q_i, \psi_i, q_{i+1}) \in \rightarrow$ such that
  $$\sigma_i \models \beta \psi_i.$$
Run or Not Run Examples

\[ q = (q_i)_{i \in \mathbb{N}_0}, \quad q_0 = q_{\text{ini}}, \]
\[ \forall i \in \mathbb{N}_0 \exists (q_i, \psi_i, q_{i+1}) \in \rightarrow : (\sigma_i, \text{cons}_i, \text{Snd}_i) \models_\beta \psi_i \]

\[
\begin{align*}
\psi: & \quad \sigma_0 \not\beta a(x, y) \quad (\Rightarrow \sigma_0 \not\beta \tau_a(x, y)) \\
& \quad \sigma_1 \not\beta a(x, y) \quad \sigma_2 \not\beta e(x) \\
& \quad \sigma_3 \not\beta b(y) \land \tau c \\
& \quad \sigma_4 \not\beta e(x) \land d(y, x) \\
& \quad \vdots \\
R_0 = & \quad q_1 q_1 q_1 q_1 q_2 q_2 q_5 q_5 q_5 q_5 \ldots \\
& \quad \vdots \end{align*}
\]

\[
\begin{array}{c}
\sigma_0 \\
\sigma_1 \\
\sigma_2 \\
\sigma_3 \\
\sigma_4 \\
\sigma_5 \\
\end{array}
\]
Definition.
We say $\mathcal{B} = (\text{Expr}_\mathcal{B}, X, Q, q_\text{ini}, \rightarrow, Q_F)$ accepts $w$ (under valuation $\beta : X \rightarrow \mathcal{D}(X)$) if and only if $\mathcal{B}$ 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 $\mathcal{L}_\beta(\mathcal{B})$ of words over $\mathcal{S}$ that are accepted by $\mathcal{B}$ under $\beta$ the language of $\mathcal{B}$.
Language of the Example TBA

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

$$(\sigma_i, Snd_i, cons_i)_{i \in \mathbb{N}_0}$$

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

- for $0 \leq i < n$, $\sigma_i \not\vDash_\beta a(x, y)$
- $\sigma_n \vDash_\beta a(x, y)$
- for $n < i < m$, $\sigma_i \not\vDash_\beta b(y)$
- $\sigma_m \vDash_\beta b(y) \land c$ and
  - for $m < i < k$, $\sigma_i \not\vDash_\beta d(y, x)$
  - $\sigma_k \vDash_\beta d(y, x)$
  - for $k < i < \ell$, $\sigma_i \not\vDash_\beta e(x)$
  - $\sigma_\ell \vDash_\beta e(x)$, or
  - $\sigma_m \vDash_\beta b(y) \land \neg c$
Back to Main Track: Live Sequence Charts Semantics
Recall Idea: Accepting Words by Advancing the Cut

Let \( w = (\sigma_i, cons_i, Snd_i)_{i \in \mathbb{N}_0} \) be a word over \( \mathcal{L} \) and \( \mathcal{D} \).

**Intuitively** (and for now **disregarding** cold conditions), an LSC body \((I, (\mathcal{L}, \preceq), \sim, \mathcal{L}, \text{Msg}, \text{Cond}, \text{LocInv})\) 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 \( i_j \leq k < i_{j+1} \), \((\sigma_k, cons_k, Snd_k)\) satisfies (under \( \beta \)) the **hold condition** of \( C_{j-1} \),
  - \((\sigma_{i_j}, cons_{i_j}, Snd_{i_j})\) satisfies (under \( \beta \)) the **transition condition** of \( F_j \),
- \( C_n \) is cold,
- for all \( i_n < k \), \((\beta_k, \mu_{i_j}, t_{i_j})\) satisfies (under \( \beta \)) the **hold condition** of \( C_n \).
**Language of LSC Body**

The **language** of the body

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

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

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

with

- \(\text{Expr}_L = \text{Expr}_\mathcal{I}(V, \mathcal{E}(\mathcal{I}))\)
- \(Q\) is the set of cuts of \((\mathcal{L}, \preceq)\), \(q_{ini}\) 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

\[ B_L = (Expr_B, X, Q, q_{ini}, \rightarrow, Q_F) \] with

- \( Expr_B = Expr_\mathcal{L}(V, \mathcal{E}(\mathcal{L})) \)
- \( Q \) is the set of cuts of \((\mathcal{L}, \preceq)\), \( q_{ini} \) 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 \((q, \psi, \mathcal{L})\).

“what allows us to stay at this cut”

“characterisation of firedset \( F_n \)”

“what allows us to legally exit”
Let $\mathcal{I} = (\mathcal{T}, \mathcal{C}, V, atr)$ be a signature and $X$ a set of logical variables.

The **signal and integer expressions** $Expr_{\mathcal{I}}(V, \mathcal{C}(\mathcal{I}))$ over $\mathcal{I}$ are defined by the grammar:

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

where $\text{expr} \in Expr_{\mathcal{I}}$, $E \in \mathcal{C}$, $x, y \in X$. 
Satisfaction of Signal and Integer Expressions

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

- \((\sigma, \text{cons}, \text{Snd}) \models_\beta \text{true}\)
- \((\sigma, \text{cons}, \text{Snd}) \models_\beta \neg \psi\) 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 \((\beta(x), (E, \vec{d}), \beta(y)) \in \text{Snd}\)
- \((\sigma, \text{cons}, \text{Snd}) \models_\beta E'_x\) if and only if \((\beta(x), (E, \vec{d})) \in \text{cons}\)

\[\text{e.g.: } \text{d.x}>5\]
Satisfaction of Signal and Integer Expressions

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

- \((\sigma, \text{cons}, \text{Snd}) \models_\beta \text{true}\)
- \((\sigma, \text{cons}, \text{Snd}) \models_\beta \neg \psi\) 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
  \(\quad (\sigma, \text{cons}, \text{Snd}) \models_\beta \psi_1\) or \((\sigma, \text{cons}, \text{Snd}) \models_\beta \psi_2\)
- \((\sigma, \text{cons}, \text{Snd}) \models_\beta \text{expr}\) if and only if \(I[\text{expr}](\sigma, \beta) = 1\)
- \((\sigma, \text{cons}, \text{Snd}) \models_\beta E^1_{x,y}\) if and only if \((\beta(x), (E, \vec{d}), \beta(y)) \in \text{Snd}\)
- \((\sigma, \text{cons}, \text{Snd}) \models_\beta E^2_x\) 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

\[
E_y^? \land \neg expr \\
E_y^? \land expr \\
F_y^! \\
F_y^! \\
F_x^? \\
F_x^? \\
true
\]
Some Helper Functions

- **Messages of a location:**
  \[ B(l) := \{ b \in B \mid \exists l' : (l, b, l') \in \text{Msg} \lor (l', b, l) \in \text{Msg} \} \]
  \[ B(\{l_1, \ldots, l_n\}) := B(l_1) \cup \cdots \cup B(l_n). \]

- **Constraints relevant at cut \( q \):**
  \[ \psi(q) = \{ \psi \mid \exists l \in q, l' \notin q \mid (l, \psi, \theta, l') \in \text{LocInv} \lor (l', \psi, \theta, l) \in \text{LocInv} \} \]

\( B(\cdot) = \{ (e, B, e'), (e, e') \} \)
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 $q'$:

  $$\psi_{\text{cold}}(q, q') = \{\psi \mid \exists l \in q' \setminus q, l' \in \mathcal{L} \mid (l, \bullet, \text{expr}, \text{cold}, l') \in \text{LocInv} \lor (l', \text{expr}, \text{cold}, l, \bullet) \in \text{LocInv} \}$$

  $$\cup \{\psi \mid \exists l \in q, l' \notin q' \mid (l, \text{expr}, \text{cold}, l') \in \text{LocInv} \lor (l', \text{expr}, \text{cold}, l) \in \text{LocInv} \}$$

  $$\cup \{\psi \mid \exists L \subseteq \mathcal{L} \mid (L, \psi, \text{cold}) \in \text{Cond} \land L \cap (q' \setminus q) \neq \emptyset \}$$
Recall: Intuition

\[ B_L = (Expr_B, X, Q, q_{ini}, \rightarrow, Q_F) \] with

- \( Expr_B = Expr_\mathcal{L}(V, \mathcal{E}(\mathcal{L})) \)
- \( Q \) is the set of cuts of \((\mathcal{L}, \preceq)\), \( q_{ini} \) 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 \((q, \psi, \mathcal{L})\).

\[ \text{true} \]

\[ \ldots \text{B and } v \not\rightarrow \]

\[ \text{“what allows us to stay at this cut”} \]

\[ \text{not B recv and not C recv} \]

\[ \text{“characterisation of firedset } F_n \” \]

\[ \text{“what allows us to legally exit”} \]

\[ : C_1 \quad : C_2 \quad : C_3 \]

\[ v = 0 \]
Loops

- How long may we **legally** stay at a cut $q$?

**Intuition:** those $(\sigma_i, cons_i, Snd_i)$ are allowed to fire the self-loop $(q, \psi, q)$ where

- $cons_i \cup Snd_i$ comprises only irrelevant messages:
  - **weak mode:** (permissive)
    - 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 $q$.

- $\psi := \neg(\bigvee B(F)) \land \bigwedge \psi(q)$. \{ **weak mode** \}
- $\psi := \neg(\bigvee B(F)) \land \bigwedge \psi(q)$.

\[ \forall x \in \{l_{i, j} : (l_{i, j}, q, l_{i, j+1}) \} \]
\[ \forall x \in \{l_{i, j} : (l_{i, j}, q, l_{i, j+1}) \} \]
\[ \forall x \in \{l_{i, j} : (l_{i, j}, q, l_{i, j+1}) \} \]

conjoin all constraints in the set $\Psi(q)$
• 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 $q \rightsquigarrow_F q'$ 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 $q \rightsquigarrow_F q'$ (unique).

\[ \psi := \bigwedge_{F \in \mathcal{F}} B(F) \land \neg \left( \bigvee_{F \in \mathcal{F}} (B(F_1) \cup \cdots \cup B(F_n)) \setminus B(F) \right) \land \bigwedge \psi(q, q'), \]

\[ \text{weak mode} \]

\[ \text{the msgrs. in firedset} \]

\[ \text{and no othr firedset} \]

\[ \text{respect conditions relvnt at } q' \]
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 $(q, \psi, L)$ for which there exists a firedset $F$ and some $q'$ such that $q \rightsquigarrow_F q'$ 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).

- $\sigma_i$ does not satisfy one cold constraint (or loc inv)

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

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

we could move from $q$ to $q'$ with firedset $F_i$. But the cold constraint doesn't hold.
Example

\[ v = 0 \]

Diagram with nodes labeled as follows:

- \( C_1 \)
- \( C_2 \)
- \( C_3 \)
- Node labeled with \( x > 3 \)
- Nodes labeled with letters: \( A, B, C, D, E \)
- Edges labeled with variables: \( l_{1,0}, l_{1,1}, l_{1,2}, l_{1,3}, l_{1,4}, l_{2,0}, l_{2,1}, l_{2,2}, l_{2,3}, l_{3,0}, l_{3,1}, l_{3,2} \)
Finally: The LSC Semantics

A full LSC $L$ consist of

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

- **universal** (= hot), **initial**, and
  
  $\forall w \in W \forall \beta : X \rightarrow \text{dom}(w_0) \bullet w$ activates $L \implies w \in \mathcal{L}(B_L)$.

- **universal** (= hot), **invariant**, and
  
  $\forall w \in W \forall k \in \mathbb{N}_0 \forall \beta : X \rightarrow \text{dom}(w_k) \bullet w/k$ activates $L \implies w/k \in \mathcal{L}(B_L)$.

- **existential** (= cold), **initial**, and
  
  $\exists w \in W \exists \beta : X \rightarrow \text{dom}(w_0) \bullet w$ activates $L \land w \in \mathcal{L}(B_L)$.

- **existential** (= cold), **invariant**, and
  
  $\exists w \in W \exists k \in \mathbb{N}_0 \exists \beta : X \rightarrow \text{dom}(w_k) \bullet w/k$ activates $L \land w/k \in \mathcal{L}(B_L)$. 
Back to UML: Interactions
Interactions as Reflective Description

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

---

**Figure 14.26** - Sequence Diagram with time and timing concepts

**Figure 14.27** - Communication diagram

**Figure 14.28** - Compact Lifeline with States

**Figure 14.29** - Timing Diagram with more than one Lifeline and with Messages

---

*OMG, 2007b* 513, 515, 522
Interactions as Reflective Description

- In UML, reflective (temporal) descriptions are subsumed by **interactions**.
- A UML model \( \mathcal{M} = (\mathcal{CD}, \mathcal{IM}, \mathcal{OD}, \mathcal{I}) \) 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).
Interactions as Reflective Description

- In UML, reflective (temporal) descriptions are subsumed by interactions.
- A UML model $\mathcal{M} = (CD, IM, OD, I)$ has a set of interactions $I$.
- An interaction $I \in I$ 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.
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


