Contents & Goals

Last Lecture:
- Started DC Syntax and Semantics: Symbols, State Assertions

This Lecture:
- **Educational Objectives:** Capabilities for following tasks/questions.
  - Read (and at best also write) Duration Calculus terms and formulae.
- **Content:**
  - Duration Calculus Formulae
  - Duration Calculus Abbreviations
  - Satisfiability, Realisability, Validity
Duration Calculus Cont’d
We will introduce three (or five) syntactical “levels”:

(i) **Symbols:**

\[ f, g, \quad true, false, =, <, >, \leq, \geq, \quad x, y, z, \quad X, Y, Z, \quad d \]

(ii) **State Assertions:**

\[ P ::= 0 \mid 1 \mid X = d \mid \neg P_1 \mid P_1 \land P_2 \]

(iii) **Terms:**

\[ \theta ::= x \mid \ell \mid \int P \mid f(\theta_1, \ldots, \theta_n) \]

(iv) **Formulae:**

\[ F ::= p(\theta_1, \ldots, \theta_n) \mid \neg F_1 \mid F_1 \land F_2 \mid \forall x \bullet F_1 \mid F_1 ; F_2 \]

(v) **Abbreviations:**

\[ [], \quad [P], \quad [P]^t, \quad [P]^{\leq t}, \quad \Diamond F, \quad \Box F \]
**Remark 2.5.** The semantics $\mathcal{I}[\theta]$ of a term is insensitive against changes of the interpretation $\mathcal{I}$ at individual time points.

**Remark 2.6.** The semantics $\mathcal{I}[\theta](\mathcal{V}, [b, e])$ of a **rigid** term does not depend on the interval $[b, e]$. 
Duration Calculus: Overview

We will introduce three (or five) syntactical “levels”:

(i) **Symbols:**

\[ f, g, \ true, false, =, <, >, \leq, \geq, \ x, y, z, \ X, Y, Z, \ d \]

(ii) **State Assertions:**

\[ P ::= 0 | 1 | X = d | \neg P_1 | P_1 \land P_2 \]

(iii) **Terms:**

\[ \theta ::= x | \ell | \int P | f(\theta_1, \ldots, \theta_n) \]

(iv) **Formulae:**

\[ F ::= p(\theta_1, \ldots, \theta_n) | \neg F_1 | F_1 \land F_2 | \forall x \bullet F_1 | F_1 ; F_2 \]

(v) **Abbreviations:**

\[ [\ ] , [P], [P]^t, [P]^{\leq t}, \diamond F, \Box F \]
The set of **DC formulae** is defined by the following grammar:

\[
F ::= p(\theta_1, \ldots, \theta_n) | \neg F_1 | F_1 \land F_2 | \forall x \bullet F_1 | F_1 ; F_2
\]

where \( p \) is a predicate symbol, \( \theta_i \) a term, \( x \) a global variable.

- **chop operator**: ‘;’
- **atomic formula**: \( p(\theta_1, \ldots, \theta_n) \)
- **rigid formula**: all terms are rigid
- **chop free**: ‘;’ doesn’t occur
- usual notion of **free** and **bound** (global) variables

Note: quantification only over (**first-order**) global variables, not over (**second-order**) state variables.
To avoid parentheses, we define the following five priority groups from highest to lowest priority:

- \( \neg \) (negation)
- \( ; \) (chop)
- \( \land, \lor \) (and/or)
- \( \Rightarrow, \Leftarrow \) (implication/equivalence)
- \( \exists, \forall \) (quantifiers)

Examples:

- \( \neg F ; F \lor H \)
- \( \forall x \bullet F \land G \)
Syntactic Substitution...

...of a term $\theta$ for a variable $x$ in a formula $F$.

- We use

$$F[x := \theta]$$

to denote the formula that results from performing the following steps:

(i) transform $F$ into $\tilde{F}$ by (consistently) renaming bound variables such that no free occurrence of $x$ in $\tilde{F}$ appears within a quantified subformula $\exists z \bullet G$ or $\forall z \bullet G$ for some $z$ occurring in $\theta$,

(ii) textually replace all free occurrences of $x$ in $\tilde{F}$ by $\theta$. 
Syntactic Substitution...

...of a term $\theta$ for a variable $x$ in a formula $F$.

- We use

$$F[x := \theta]$$

...to denote the formula that results from performing the following steps:

- transform $F$ into $\tilde{F}$ by (consistently) renaming bound variables such that no free occurrence of $x$ in $\tilde{F}$ appears within a quantified subformula $\exists z \cdot G$ or $\forall z \cdot G$ for some $z$ occurring in $\theta$,
- textually replace all free occurrences of $x$ in $\tilde{F}$ by $\theta$.

**Examples:**

$$F := (x \geq y \implies \exists z \cdot z \geq 0 \land x = y + z), \quad \theta_1 := \ell, \quad \theta_2 := \ell + z,$$

- $$F[x := \theta_1] = (x \geq y \implies \exists z \cdot z \geq 0 \land x = y + z)$$
- $$F[x := \theta_2] = (x \geq y \implies \exists z \cdot z \geq 0 \land x = y + z)$$
The **semantics** of a **formula** is a function

\[
\mathcal{I}[F] : \text{Val} \times \text{Intv} \rightarrow \{\text{tt}, \text{ff}\}
\]

i.e. \(\mathcal{I}[F](\mathcal{V}, [b, e])\) is the truth value of \(F\) under interpretation \(\mathcal{I}\) and valuation \(\mathcal{V}\) in the interval \([b, e]\).

This value is defined **inductively** on the structure of \(F\):

\[
\mathcal{I}[p(\theta_1, \ldots, \theta_n)](\mathcal{V}, [b, e]) = \hat{p}(\mathcal{I}[\theta_1](\mathcal{V}, [b, e]), \ldots, \mathcal{I}[\theta_n](\mathcal{V}, [b, e])),
\]

\[
\mathcal{I}[\neg F_1](\mathcal{V}, [b, e]) = \text{tt} \text{ iff } \mathcal{I}[F_1](\mathcal{V}, [b, e]) = \text{ff},
\]

\[
\mathcal{I}[F_1 \land F_2](\mathcal{V}, [b, e]) = \text{tt} \text{ iff } \mathcal{I}[F_1](\mathcal{V}, [b, e]) = \mathcal{I}[F_2](\mathcal{V}, [b, e]) = \text{tt},
\]

\[
\mathcal{I}[\forall x \bullet F_1](\mathcal{V}, [b, e]) = \text{tt} \text{ iff for all } a \in \mathbb{R}, \mathcal{I}[F_1[x := a]](\mathcal{V}, [b, e]) = \text{tt}
\]

\[
\mathcal{I}[F_1; F_2](\mathcal{V}, [b, e]) = \text{iff} \text{ there is an } m \in [b, e] \text{ such that } \mathcal{I}[F_1](\mathcal{V}, [b, m]) = \mathcal{I}[F_2](\mathcal{V}, [m, e]) = \text{tt}.
\]
Formulae: Example

\[ F := \int L = 0 ; \int L = 1 \]

- \[ \mathcal{L}[F](V, [0, 2]) = \]
Remark 2.10. [Rigid and chop-free] Let $F$ be a duration formula, $\mathcal{I}$ an interpretation, $\mathcal{V}$ a valuation, and $[b, e] \in \text{Intv}$.

- If $F$ is **rigid**, then
  $$\forall [b', e'] \in \text{Intv} : \mathcal{I}[F](\mathcal{V}, [b, e]) = \mathcal{I}[F](\mathcal{V}, [b', e']).$$

- If $F$ is **chop-free** or $\theta$ is **rigid**, then in the calculation of the semantics of $F$, every occurrence of $\theta$ denotes the same value.
Lemma 2.11. [Substitution]
Consider a formula $F$, a global variable $x$, and a term $\theta$ such that $F$ is chop-free or $\theta$ is rigid.

Then for all interpretations $\mathcal{I}$, valuations $\mathcal{V}$, and intervals $[b, e]$,

$$\mathcal{I}[F[x := \theta]](\mathcal{V}, [b, e]) = \mathcal{I}[F](\mathcal{V}[x := a], [b, e])$$

where $a = \mathcal{I}[\theta](\mathcal{V}, [b, e])$.

- $F := \ell = x ; \ell = x \implies \ell = 2 \cdot x, \quad \theta := \ell$
Duration Calculus: Overview

We will introduce three (or five) syntactical “levels”:

(i) **Symbols:**

\[ f, g, \quad true, false, =, <, >, \leq, \geq, \quad x, y, z, \quad X, Y, Z, \quad d \]

(ii) **State Assertions:**

\[ P ::= 0 \mid 1 \mid X = d \mid \neg P_1 \mid P_1 \land P_2 \]

(iii) **Terms:**

\[ \theta ::= x \mid \ell \mid \int P \mid f(\theta_1, \ldots, \theta_n) \]

(iv) **Formulae:**

\[ F ::= p(\theta_1, \ldots, \theta_n) \mid \neg F_1 \mid F_1 \land F_2 \mid \forall x \bullet F_1 \mid F_1 ; F_2 \]

(v) **Abbreviations:**

\[ [\ ] \mid [P] \mid [P]^t \mid [P]^{\leq t} \mid \Diamond F \mid \square F \]
Duration Calculus Abbreviations
Abbreviations

- \([\mathcal{N}] := \ell = 0\) (point interval)
- \([P] := \int P = \ell \land \ell > 0\) (almost everywhere)
- \([P]^t := [P] \land \ell = t\) (for time \(t\))
- \([P]^{\leq t} := [P] \land \ell \leq t\) (up to time \(t\))
- \(\diamond F := \text{true} ; F ; \text{true}\) (for some subinterval)
- \(\Box F := \neg \diamond \neg F\) (for all subintervals)
Abbreviations: Examples

\[ \mathcal{T} \left[ \int L = 0 \right] = \mathcal{I} \left( \mathcal{V}, [0, 2] \right) = \]

\[ \mathcal{T} \left[ \int L = 1 \right] = \mathcal{I} \left( \mathcal{V}, [2, 6] \right) = \]

\[ \mathcal{T} \left[ \int L = 0 ; \int L = 1 \right] = \mathcal{I} \left( \mathcal{V}, [0, 6] \right) = \]

\[ \mathcal{T} \left[ \lnot L \right] = \mathcal{I} \left( \mathcal{V}, [0, 2] \right) = \]

\[ \mathcal{T} \left[ L \right] = \mathcal{I} \left( \mathcal{V}, [2, 3] \right) = \]

\[ \mathcal{T} \left[ \lnot L ; \lnot L \right] = \mathcal{I} \left( \mathcal{V}, [0, 3] \right) = \]

\[ \mathcal{T} \left[ \lnot L ; L ; \lnot L \right] = \mathcal{I} \left( \mathcal{V}, [0, 6] \right) = \]

\[ \mathcal{T} \left[ \lnot L \right] = \mathcal{I} \left( \mathcal{V}, [0, 6] \right) = \]

\[ \mathcal{T} \left[ \lnot L \right] = \mathcal{I} \left( \mathcal{V}, [0, 6] \right) = \]

\[ \mathcal{T} \left[ \lnot L \right] = \mathcal{I} \left( \mathcal{V}, [0, 6] \right) = \]

\[ \mathcal{T} \left[ \lnot L \right] = \mathcal{I} \left( \mathcal{V}, [0, 6] \right) = \]

\[ \mathcal{T} \left[ \lnot L \right] = \mathcal{I} \left( \mathcal{V}, [0, 6] \right) = \]

\[ \mathcal{T} \left[ \lnot L \right] = \mathcal{I} \left( \mathcal{V}, [0, 6] \right) = \]
Duration Calculus: Preview

- Duration Calculus is an **interval logic**.
- Formulae are evaluated in an **(implicitly given)** interval.

**Strangest operators:**

- **almost everywhere** — Example: \([G]\)
  (Holds in a given interval \([b, e]\) iff the gas valve is open almost everywhere.)

- **chop** — Example: \((\lceil \neg I \rceil; \lceil I \rceil; \lceil \neg I \rceil) \implies \ell \geq 1\)
  (Ignition phases last at least one time unit.)

- **integral** — Example: \(\ell \geq 60 \implies \int L \leq \frac{\ell}{20}\)
  (At most 5% leakage time within intervals of at least 60 time units.)

- \(G, F, I, H : \{0, 1\}\)
- Define \(L : \{0, 1\}\) as \(G \land \neg F\).
DC Validity, Satisfiability, Realisability
Let $\mathcal{I}$ be an interpretation, $\mathcal{V}$ a valuation, $[b, e]$ an interval, and $F$ a DC formula.

- $\mathcal{I}, \mathcal{V}, [b, e] \models F$ ("$F$ holds in $\mathcal{I}, \mathcal{V}, [b, e]$") iff $\mathcal{I}[F](\mathcal{V}, [b, e]) = \text{tt}$. 
Validity, Satisfiability, Realisability

Let $\mathcal{I}$ be an interpretation, $\mathcal{V}$ a valuation, $[b, e]$ an interval, and $F$ a DC formula.

- $\mathcal{I}, \mathcal{V}, [b, e] \models F$ ("$F$ holds in $\mathcal{I}, \mathcal{V}, [b, e]$"") iff $\mathcal{I}[F](\mathcal{V}, [b, e]) = \text{tt}$.

- $F$ is called **satisfiable** iff it holds in some $\mathcal{I}, \mathcal{V}, [b, e]$. 
Validity, Satisfiability, Realisability

Let $\mathcal{I}$ be an interpretation, $\mathcal{V}$ a valuation, $[b, e]$ an interval, and $F$ a DC formula.

- $\mathcal{I}, \mathcal{V}, [b, e] \models F$ ("$F$ holds in $\mathcal{I}$, $\mathcal{V}$, $[b, e]$") iff \[ \mathcal{I}[F](\mathcal{V}, [b, e]) = \text{tt}. \]

- $F$ is called **satisfiable** iff it holds in some $\mathcal{I}, \mathcal{V}, [b, e]$.

- $\mathcal{I}, \mathcal{V} \models F$ ("$\mathcal{I}$ and $\mathcal{V}$ realise $F$") iff \[ \forall [b, e] \in \text{Intv} : \mathcal{I}, \mathcal{V}, [b, e] \models F. \]
Validity, Satisfiability, Realisability

Let $\mathcal{I}$ be an interpretation, $\mathcal{V}$ a valuation, $[b, e]$ an interval, and $F$ a DC formula.

- $\mathcal{I}, \mathcal{V}, [b, e] \models F$ ("$F$ holds in $\mathcal{I}, \mathcal{V}, [b, e]$") iff $\mathcal{I}[F](\mathcal{V}, [b, e]) = \text{tt}$.

- $F$ is called **satisfiable** iff it holds in some $\mathcal{I}, \mathcal{V}, [b, e]$.

- $\mathcal{I}, \mathcal{V} \models F$ ("$\mathcal{I}$ and $\mathcal{V}$ realise $F$"") iff $\forall [b, e] \in \text{Intv} : \mathcal{I}, \mathcal{V}, [b, e] \models F$.

- $F$ is called **realisable** iff some $\mathcal{I}$ and $\mathcal{V}$ realise $F$. 
Validity, Satisfiability, Realisability

Let $\mathcal{I}$ be an interpretation, $\mathcal{V}$ a valuation, $[b, e]$ an interval, and $F$ a DC formula.

- $\mathcal{I}, \mathcal{V}, [b, e] \models F$ ("$F$ holds in $\mathcal{I}, \mathcal{V}, [b, e]$") iff $\mathcal{I}[F](\mathcal{V}, [b, e]) = \text{tt}$. 

- $F$ is called **satisfiable** iff it holds in some $\mathcal{I}, \mathcal{V}, [b, e]$.

- $\mathcal{I}, \mathcal{V} \models F$ ("$\mathcal{I}$ and $\mathcal{V}$ realise $F$") iff $\forall [b, e] \in \text{Intv} : \mathcal{I}, \mathcal{V}, [b, e] \models F$.

- $F$ is called **realisable** iff some $\mathcal{I}$ and $\mathcal{V}$ realise $F$.

- $\mathcal{I} \models F$ ("$\mathcal{I}$ realises $F$") iff $\forall \mathcal{V} \in \text{Val} : \mathcal{I}, \mathcal{V} \models F$. 

Validity, Satisfiability, Realisability

Let $\mathcal{I}$ be an interpretation, $\mathcal{V}$ a valuation, $[b, e]$ an interval, and $F$ a DC formula.

- $\mathcal{I}, \mathcal{V}, [b, e] \models F$ ("$F$ holds in $\mathcal{I}, \mathcal{V}, [b, e]$") iff $\mathcal{I}[F](\mathcal{V}, [b, e]) = \text{tt}$. 

- $F$ is called **satisfiable** iff it holds in some $\mathcal{I}, \mathcal{V}, [b, e]$.

- $\mathcal{I}, \mathcal{V} \models F$ ("$\mathcal{I}$ and $\mathcal{V}$ realise $F$") iff $\forall [b, e] \in \text{Intv} : \mathcal{I}, \mathcal{V}, [b, e] \models F$. 

- $F$ is called **realisable** iff some $\mathcal{I}$ and $\mathcal{V}$ realise $F$.

- $\mathcal{I} \models F$ ("$\mathcal{I}$ realises $F$") iff $\forall \mathcal{V} \in \text{Val} : \mathcal{I}, \mathcal{V} \models F$. 

- $\models F$ ("$F$ is valid") iff $\forall$ interpretation $\mathcal{I} : \mathcal{I} \models F$. 

Remark 2.13. For all DC formulae $F$,

- $F$ is satisfiable iff $\neg F$ is not valid,
  $F$ is valid iff $\neg F$ is not satisfiable.
- If $F$ is valid then $F$ is realisable, but not vice versa.
- If $F$ is realisable then $F$ is satisfiable, but not vice versa.
Examples: Valid? Realisable? Satisfiable?

- $\ell \geq 0$
- $\ell = \int 1$
- $\ell = 30 \iff \ell = 10 ; \ell = 20$
- $((F ; G) ; H) \iff (F ; (G ; H))$
- $\int L \leq x$
- $\ell = 2$
Initial Values

- \( \mathcal{I}, \mathcal{V} \models_0 F \) ("\( \mathcal{I} \) and \( \mathcal{V} \) realise \( F \) from 0") iff
  \[ \forall t \in \text{Time} : \mathcal{I}, \mathcal{V}, [0, t] \models F. \]

- \( F \) is called **realisable from 0** iff some \( \mathcal{I} \) and \( \mathcal{V} \) realise \( F \) from 0.

- Intervals of the form \([0, t]\) are called **initial intervals**.

- \( \mathcal{I} \models_0 F \) ("\( \mathcal{I} \) realises \( F \) from 0") iff
  \[ \forall \mathcal{V} \in \text{Val} : \mathcal{I}, \mathcal{V} \models_0 F. \]

- \( \models_0 F \) ("\( F \) is **valid from 0**") iff
  \[ \forall \text{ interpretation } \mathcal{I} : \mathcal{I} \models_0 F. \]
For all interpretations $\mathcal{I}$, valuations $\mathcal{V}$, and DC formulae $F$,

(i) $\mathcal{I}, \mathcal{V} \models F$ implies $\mathcal{I}, \mathcal{V} \models_0 F$,

(ii) if $F$ is realisable then $F$ is realisable from 0, but not vice versa,

(iii) $F$ is valid iff $F$ is valid from 0.
Specification and Semantics-based Correctness Proofs of Real-Time Systems with DC
(i) Choose a collection of observables ‘Obs’.
(ii) Provide the requirement/specification ‘Spec’ as a conjunction of DC formulae (over ‘Obs’).
(iii) Provide a description ‘Ctrl’ of the controller in form of a DC formula (over ‘Obs’).
(iv) We say ‘Ctrl’ is correct (wrt. ‘Spec’) iff

$$\models_0 \text{Ctrl} \implies \text{Spec}.$$
(i) Choose **observables**:

- two boolean observables \( G \) and \( F \)
  
  (i.e. \( \text{Obs} = \{G, F\}, \mathcal{D}(G) = \mathcal{D}(F) = \{0, 1\} \))

- \( G = 1 \): gas valve open
- \( F = 1 \): have flame
- define \( L := G \land \neg F \) (leakage)

(ii) Provide the **requirement**:

\[
\text{Req :} \iff \Box(\ell \geq 60 \implies 20 \cdot \int L \leq \ell)
\]
(iii) Provide a description ‘Ctrl’ of the **controller** in form of a DC formula (over ‘Obs’). Here, firstly consider a **design**:

- **Des-1**: $\iff \Box([L] \implies \ell \leq 1)$

- **Des-2**: $\iff \Box([L] ; [\neg L] ; [L] \implies \ell > 30)$

(iv) Prove **correctness**:

- We want (or do we want $|=_{0\ldots?}$):

  $|= (\text{Des-1} \land \text{Des-2} \implies \text{Req})$  
  
  (Thm. 2.16)
(iii) Provide a description ‘Ctrl’ of the controller in form of a DC formula (over ‘Obs’). Here, firstly consider a design:

- Des-1: $\square([L] \implies \ell \leq 1)$
- Des-2: $\square([L] ; [\neg L] ; [L] \implies \ell > 30)$

(iv) Prove correctness:

- We want (or do we want $|=0...?$):

$$|= (\text{Des-1} \land \text{Des-2} \implies \text{Req})$$

(Thm. 2.16)

- We do show

$$|= \text{Req-1} \implies \text{Req}$$

(Lem. 2.17)

with the simplified requirement

$$\text{Req-1} := \square(\ell \leq 30 \implies \int L \leq 1),$$
Claim:

\[ \vdash \Box(\ell \leq 30 \implies \int L \leq 1) \implies \Box(\ell \geq 60 \implies 20 \cdot \int L \leq \ell) \]

Proof:
Claim:

\[ \vdash \square(\ell \leq 30 \implies \int L \leq 1) \implies \square(\ell \geq 60 \implies 20 \cdot \int L \leq \ell) \]

Proof:

- Assume ‘Req-1’.
Claim:

\[\models \square(\ell \leq 30 \implies \int L \leq 1) \implies \square(\ell \geq 60 \implies 20 \cdot \int L \leq \ell)\]

Proof:

- Assume ‘Req-1’.
- Let \( L_\mathcal{I} \) be any interpretation of \( L \), and \([b, e]\) an interval with \( e - b \geq 60 \).
Gas Burner Revisited: Lemma 2.17

Claim:

\[ \models \square(\ell \leq 30 \implies \int L \leq 1) \implies \square(\ell \geq 60 \implies 20 \cdot \int L \leq \ell) \]

Proof:

- Assume ‘Req-1’.
- Let \( L_\mathcal{I} \) be any interpretation of \( L \), and \([b, e]\) an interval with \( e - b \geq 60 \).
- Show “\( 20 \cdot \int L \leq \ell \)”, i.e.

\[ \mathcal{I}[20 \cdot \int L \leq \ell](\mathcal{V}, [b, e]) = \text{tt} \]

i.e.

\[ 20 \cdot \int_{b}^{e} L_\mathcal{I}(t) \, dt \leq (e - b) \]
Set \( n := \left\lceil \frac{e-b}{30} \right\rceil \), i.e. \( n \in \mathbb{N} \) with \( n - 1 < \frac{e-b}{30} \leq n \), and split the interval \( b \) to \( e \):

\[
\begin{align*}
&b + 30 & b + 60 & b + 30(n - 2) & b + 30(n - 1) & b + 30n \\
&--- & | & \cdots & | & ---
\end{align*}
\]
Theorem 2.18.  
For all state assertions $P$ and all real numbers $r_1, r_2 \in \mathbb{R}$,

(i) $\models \int P \leq \ell$,
(ii) $\models (\int P = r_1) ; (\int P = r_2) \implies \int P = r_1 + r_2$,
(iii) $\models [\neg P] \implies \int P = 0$,
(iv) $\models [\emptyset] \implies \int P = 0$. 
Claim:

$$\models (\blacksquare([L] \implies \ell \leq 1) \land \blacksquare([L] ; [\neg L] ; [L] \implies \ell > 30)) \implies \blacksquare(\ell \leq 30 \implies \int L \leq 1)$$

Proof:
Claim:

\[ \models (\square(\neg L \implies \ell \leq 1) \land \square([L] \land [\neg L] \land [L \implies \ell > 30])) \implies \square(\ell \leq 30 \implies \int L \leq 1) \]

Proof:

(i) \models \int P \leq \ell, \quad (iv) \models \square \implies \int P = 0,

(ii) \models (\int P = r_1) \land (\int P = r_2) \implies \int P = r_1 + r_2,

(iii) \models \neg P \implies \int P = 0,
Gas Burner Revisited: Lemma 2.18
References