Real-Time Systems

Lecture 3: Duration Calculus I

2017-10-26

Dr. Bernd Westphal

Albert-Ludwigs-Universität Freiburg, Germany

Content

Introduction

- **Observables and Evolutions** ✓
- **Duration Calculus** (DC)
- Semantical Correctness Proofs
- DC Decidability
- DC Implementables
- **PLC-Automata**

- **Timed Automata** (TA), Uppaal
- Networks of Timed Automata
- Region/Zone-Abstraction
- TA model-checking
- Extended Timed Automata
- Undecidability Results

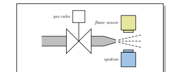
$obs : \text{Time} \to \mathscr{D}(obs)$

$\langle obs_0, \nu_0\rangle, t_0 \xrightarrow{\lambda_0} \langle obs_1, \nu_1\rangle, t_1 \ldots$

- **Automatic Verification**...
  ...whether a TA satisfies a DC formula, observer-based
- **Recent Results**:
  - **Timed Sequence Diagrams**, or **Quasi-equal Clocks**, or **Automatic Code Generation**, or ...

Duration Calculus: Preview

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



- $G, F, I, H : \{0,1\}$
- Define $L : \{0,1\}$ as $G \land \neg F$.

Strangest operators: $\lceil \text{true} \rceil$

- **almost everywhere** – Example: $\lceil G \rceil$
  (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.)

$\lceil \neg I \rceil ; \lceil I \rceil ; \lceil \neg I \rceil$ $\ell \geq 1$

Content

- **Symbols**
  - **predicate** and **function symbols**
  - **state variables** and **domain values**
  - **global** (or logical) **variables**
- **State Assertions**
  - **syntax**
  - **semantics**
- **Terms**
  - **syntax**
  - **rigid terms**
  - **intervals**
  - **semantics**
  - **remarks**

Duration Calculus: Syntax Overview

Duration Calculus: Overview

We will introduce **four syntactical categories** (and **abbreviations**):

(i) **Symbols:**
$$\overbrace{true, false, =, <, >, \leq, \geq}^{p,q} \quad f, g, \quad X, Y, Z, \quad d, \quad x, y, z,$$

(ii) **State Assertions:**
$$P ::= 0 \mid 1 \mid X = d \mid \neg P_1 \mid P_1 \land P_2$$

(iii) **Terms:**
$$\theta ::= x \mid \ell \mid \textstyle\int P \mid f(\theta_1, \ldots, \theta_n)$$

(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:**
$$\lceil\,\rceil, \quad \lceil P \rceil, \quad \lceil P \rceil^t, \quad \lceil P \rceil^{\leq t}, \quad \Diamond F, \quad \Box F$$

Duration Calculus: Symbols

Symbols: Predicate Symbols

$\overbrace{true, false, =, <, >, \leq, \geq}^{p,q}, \quad f, g, \quad X, Y, Z, \quad d, \quad x, y, z,$

Syntax

- We assume a set of **predicate symbols** to be given, typical elements p, q.
  - Each **predicate symbol** p has an **arity** $n \in \mathbb{N}_0$; shorthand notation: p/n.
  - A **predicate symbol** p/n is called a **constant** if and only if $n = 0$.
- In the following, we assume the following **predicate symbols**:
  - **constants**: $true, false$.
  - **binary** (i.e. $n = 2$): $=, <, >, \leq, \geq$.

Semantics (meaning)

- **Semantical domains**: **truth values** $\mathbb{B} = \{\mathrm{tt}, \mathrm{ff}\}$, and **real numbers** $\mathbb{R}$.
- The **semantics** of an n-ary **predicate symbol** p is a **function** from $\mathbb{R}^n$ to $\mathbb{B}$, denoted $\hat{p}$, i.e. $\hat{p} : \mathbb{R}^n \to \mathbb{B}$.
- For constants (arity $n = 0$) we have $\hat{p} \in \mathbb{B}$.
- **Examples**:
  - $\hat{true} = \mathrm{tt}$, $\hat{false} = \mathrm{ff}$,
  - $\hat{=} : \mathbb{R} \times \mathbb{R} \to \mathbb{B}$, $\hat{=}(a, b) = \mathrm{tt}$, iff $a = b$, $\hat{=}(a, b) = \mathrm{ff}$, iff $a \neq b$.
    $\hat{=}(3, 17) = \mathrm{ff}$, $\hat{=}(2, 2) = \mathrm{tt}$.

Once Again: Syntax vs. Semantics

- **Predicate symbols** are principally **freely chosen**, we could also consider the following ones:
  - $\heartsuit/1$
  - ❀/3
  - geq/2

DC symbol / syntax

- To semantically work with a **predicate symbol**, we need to define a **meaning**. One possible choice:
  - $\hat{\heartsuit} : \mathbb{R} \to \mathbb{B}$

    $\hat{\heartsuit}(a) = \begin{cases} \mathrm{tt} & \text{, if } a \in \mathbb{N} \text{ and digit sum of } a \text{ equals } 27 \\ \mathrm{ff} & \text{, otherwise} \end{cases}$
  - $\hat{❀} : \mathbb{R} \times \mathbb{R} \times \mathbb{R} \to \mathbb{B}$

    $\hat{❀}(a, b, c) = \begin{cases} \mathrm{tt} & \text{, if } ax^2 + bx + c = 0 \text{ has at least one solution} \\ \mathrm{ff} & \text{, otherwise} \end{cases}$
  - $\hat{\mathrm{geq}} : \mathbb{R} \times \mathbb{R} \to \mathbb{B}$

    $\hat{\mathrm{geq}}(a, b) = \begin{cases} \mathrm{tt} & \text{, if } a \geq b \\ \mathrm{ff} & \text{, otherwise} \end{cases}$

math. / semantics

Same Game: Function Symbols

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

Syntax

- We assume a set of **function symbols** to be given, typical elements f, g.
  - Each **function symbol** f has an **arity** $n \in \mathbb{N}_0$; shorthand notation: f/n.
  - A **function symbol** f/n is called a **constant** if and only if $n = 0$.
- In the following, we assume the following **function symbols**:
  - **constants**: $i/0$ for each $i \in \cancel{\mathbb{N}_0}, \mathbb{R}$ (for each real number from $\mathbb{R}$ we assume one function symbol)
  - **binary** (i.e. $n = 2$): $+$, $\cdot$.

Semantics (meaning)

- The **semantics** of an n-ary **function symbol** f is a **function** from $\mathbb{R}^n$ to $\mathbb{R}$, denoted $\hat{f}$, i.e. $\hat{f} : \mathbb{R}^n \to \mathbb{R}$.
- For constants (arity $n = 0$) we have $\hat{f} \in \mathbb{R}$.
- **Examples**:
  - $\hat{0} = 0 \in \mathbb{R}$, $\hat{27} = 27 \in \mathbb{R}$,
  - $\hat{+} : \mathbb{R} \times \mathbb{R} \to \mathbb{R}$, $\hat{+}(a, b) = a + b$,
    $\hat{+}(1, 2) = 3$.

One More Time

To better distinguish **syntax** from **semantics**,
we could choose to work with the following symbols for natural numbers:

- **Syntax**:
  - `zero`, `one`, `two`, …, `twentyseven`, …

  (all with arity 0)
- **Semantics**:
  - $\hat{\texttt{zero}} = 0 \in \mathbb{R}$,
  - $\hat{\texttt{one}} = 1 \in \mathbb{R}$,
  - $\hat{\texttt{two}} = 2 \in \mathbb{R}$,
  - …,
  - $\hat{\texttt{twentyseven}} = 27 \in \mathbb{R}$,
  - …

One More Time

To better distinguish **syntax** from **semantics**,
we could choose to work with the following symbols for natural numbers:

- **Syntax**:
  - $0, 1, 2, \ldots, 27, \ldots$

  (all with arity 0)
- **Semantics**:
  - $\hat{0} = 0 \in \mathbb{R}$,
  - $\hat{1} = 1 \in \mathbb{R}$,
  - $\hat{2} = 2 \in \mathbb{R}$,
  - …,
  - $\hat{27} = 27 \in \mathbb{R}$,
  - …

Symbols: State Variables and Domain Values

$$true, false, =, <, >, \leq, \geq, \quad f, g, \quad \boxed{X, Y, Z}, \quad \boxed{d}, \quad x, y, z,$$

- We assume a set 'Obs' of **state variables** or **observables**, typical elements X, Y, Z.
  - Each **state variable** X has a **finite** (semantical) **domain** $\mathcal{D}(X) = \{d_1, \dots, d_n\}$.
  - A **state variable** with domain $\{0, 1\}$ is called **boolean observable**.
- For each domain $\{d_1, \dots, d_n\}$ of a state variable in 'Obs' we assume
  - **symbols** $d_1, \dots, d_n$
  - with $\hat{d_i} = d_i, 1 \leq i \leq n$.
- **Example**:
  - state variable F ("flame sensor"), domain $\mathcal{D}(F) = \{0, 1\}$, symbols $0, 1$ with $\hat{0} = 0 \in \mathbb{N}_0$, $\hat{1} = 1 \in \mathbb{N}_0$.
  - state variable T ("traffic lights"), domain $\mathcal{D}(T) = \{\mathtt{red}, \mathtt{green}\}$, symbols $\mathtt{red}$, $\mathtt{green}$ with with $\hat{\mathtt{red}} = \mathtt{red} \in \mathcal{D}(T)$, $\hat{\mathtt{green}} = \mathtt{green} \in \mathcal{D}(T)$.

Interpretation of State Variables

- The last **semantical domain** we consider is
  - the set Time of **points in time**,
  - mostly, Time $= \mathbb{R}_0^+$ (**continuous** / **dense**), sometimes Time $= \mathbb{N}_0$ (**discrete time**).
- The **semantics** of a **state variable** is **time-dependent**.

  It is given by an **interpretation** $\mathcal{I}$, i.e. a mapping

  $$\mathcal{I} : \mathsf{Obs} \to (\mathsf{Time} \to \mathcal{D}), \qquad \mathcal{D} = \bigcup_{X \in \mathsf{Obs}} \mathcal{D}(X),$$

  assigning to each **state variable** $X \in$ Obs a function

  $$\mathcal{I}(X) : \mathsf{Time} \to \mathcal{D}(X)$$

  such that $\mathcal{I}(X)(t) \in \mathcal{D}(X)$ denotes the value that X has at time $t \in$ Time.
- For convenience, we shall **abbreviate** $\mathcal{I}(X)$ to $X_{\mathcal{I}}$.

Evolutions over Time vs. Interpretation of State Variables

- Let Obs $= \{obs_1, \dots, obs_n\}$ be a set of state variables.
- **Evolution** (over time) of Obs:

  $$\pi : \mathsf{Time} \to \mathcal{D}(obs_1) \times \dots \times \mathcal{D}(obs_n).$$

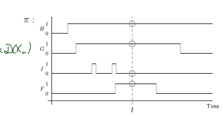
- **Interpretation** of Obs:

  $$\mathcal{I} : \mathsf{Obs} \to (\mathsf{Time} \to \mathcal{D}).$$

- Both, π and $\mathcal{I}$, represent **the same timed behaviour** if,
  - for all $t \in$ Time,
    - $\mathcal{I}(obs_i)(t) = \pi(t) \downarrow i, \quad 1 \leq i \leq n$, or
    - $\pi(t) = (\underbrace{\mathcal{I}(obs_1)(t)}, \dots, \underbrace{\mathcal{I}(obs_n)(t)}) = (obs_{1\mathcal{I}}(t), \dots, obs_{n\mathcal{I}}(t))$.

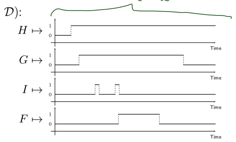
Example: Evolutions vs. Interpretation of State Variables

π: Time $\to \mathcal{D}(X_1) \times \dots \times \mathcal{D}(X_n)$



- $obs_1 = H$, $obs_2 = G$, $obs_3 = I$, $obs_3 = F$
- $\pi(t) = (1, 1, 0, 1)$, $\quad \mathcal{I}(H)(t) = H_{\mathcal{I}}(t) = \pi(t) \downarrow 1 = 1$, $\mathcal{I}(I)(t) = I_{\mathcal{I}}(t) = \pi(t) \downarrow 3 = 0$.
- $\mathcal{I}$: Obs $\to$ (Time $\to \mathcal{D}$):

Time $\to \mathcal{D}$



Predicate / Function Symbols vs. State Variables

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

Note:

- The choice of **function and predicate symbols** introduced earlier, i.e.
  - $true, false, =, <, >, \leq, \geq$,
  - $0, 1, \dots$,
  - $+, \cdot$

  and their **semantics**, i.e.
  - $\hat{true}$ is the truth value $\mathrm{tt} \in \mathbb{B}$,
  - $\hat{=} : \mathbb{R}^2 \to \mathbb{B}$ is the **equality** relation on real numbers,
  - $\hat{0}$ is the (real) number **zero** from $\mathbb{R}$,
  - $\hat{+} : \mathbb{R}^2 \to \mathbb{R}$ is the **addition function** on real numbers,

  is **fixed throughout the lecture**.
- The choice of **observables** and their **domains** **depends on the system we want to describe.**

Symbols: Global Variables

$$true, false, =, <, >, \leq, \geq, \quad f, g, \quad X, Y, Z, \quad d, \quad \boxed{x, y, z},$$

- We assume a set 'GVar' of **global** (or logical) **variables**, typical elements x, y, z.
- The semantics of a **global variable** is given by a **valuation**, i.e. a mapping

  $$\mathcal{V} : \mathsf{GVar} \to \mathbb{R}$$

  assigning to each global variable $x \in$ GVar a real number $\mathcal{V}(x) \in \mathbb{R}$.

  We use Val to denote the set of all valuations, i.e. Val $= (\mathsf{GVar} \to \mathbb{R})$.

  Global variables are **fixed over time** in system evolutions.

  $GV_{ar} = \{x, y\}$

  $\mathcal{V}_1 = \{x \mapsto 0, \ y \mapsto 1\}$

  $\mathcal{V}_2 = \{x \mapsto 3.14, \ y \mapsto 27\}$

Symbols: Overview

Syntax	Semantics (meaning)
predicate symbols	
$true, false, =, <, >, \leq, \geq$	$\hat{true} = \mathrm{tt} \in \mathbb{B}$, $\hat{=} : \mathbb{R}^2 \to \mathbb{B}$
function symbols	
$f/n, g$	$\hat{f} : \mathbb{R}^n \to \mathbb{R}$
state variables	
X, Y, Z	$\mathcal{I}(X) : \mathsf{Time} \to \mathcal{D}(X)$
domain values	
d	$\hat{d} \in \mathcal{D}(X)$
global variables	
x, y, z	$\mathcal{V}(x) \in \mathbb{R}$

Duration Calculus: State Assertions

Duration Calculus: Overview

We will introduce **four syntactical categories** (and **abbreviations**):

(i) **Symbols:** ✓

$$\overbrace{true, false, =, <, >, \leq, \geq}^{p,q}, \quad f, g, \quad X, Y, Z, \quad d, \quad x, y, z,$$

(ii) **State Assertions:**

$$P ::= 0 \mid 1 \mid X = d \mid \neg P_1 \mid P_1 \wedge P_2 \quad \vdots\ (P)$$

(iii) **Terms:**

$$\theta ::= x \mid \ell \mid \textstyle\int P \mid f(\theta_1, \dots, \theta_n) \quad \vdots\ (\theta)$$

(iv) **Formulae:**

$$F ::= p(\theta_1, \dots, \theta_n) \mid \neg F_1 \mid F_1 \wedge F_2 \mid \forall x \bullet F_1 \mid F_1 ; F_2 \quad \vdots\ (F)$$

(v) **Abbreviations:**

$$\lceil\rceil, \quad \lceil P \rceil, \quad \lceil P \rceil^t, \quad \lceil P \rceil^{\leq t}, \quad \Diamond F, \quad \Box F$$

State Assertions: Syntax

- The set of **state assertions** is defined by the following grammar:

$$P ::= \mathbf{0} \mid \mathbf{1} \mid X \mathbin{\mathbf{=}} d \mid \mathbf{\neg} P_1 \mid P_1 \mathbin{\mathbf{\wedge}} P_2$$

where
  - $X \in$ Obs is a state variable,
  - d denotes a value from X's domain.

We shall use P, Q, R to denote state assertions.

- Here, '0', '1', '=', '¬', and '∧' are like **keywords** (or terminal symbols) in programming languages.

- **Abbreviations**:
  - We shall write X instead of $\underbrace{X = 1}$ if X is **boolean**, i.e. if $\mathcal{D}(X) = \{0, 1\}$.
  - Assume the **usual precedence**: ¬ binds stronger than ∧
  - Define $\vee$, $\implies$, $\iff$ as usual.

State Assertions: Examples

$$P ::= \overset{①}{0} \mid \overset{②}{1} \mid \overset{③}{X = d} \mid \overset{④}{\neg P_1} \mid \overset{⑤}{P_1 \wedge P_2}$$

Observables $F, G, \mathcal{D}(F) = \{0, 1\}, \mathcal{D}(G) = \{0, 1, 2\}$.

$X,\ \mathcal{D}(X) = \{\boxed{F=0}\}$] state var. / dom. value

$X \hat{=} \boxed{F=0}$] state assertion

- 0 ✓ ①
- $F = 1$ ✓ ③
- F ✓ ③ + abbrev.
- $\neg(F = 1)$ ✓ ④, ③
- $\neg F$ ✓ ④ + abbr
- G ✗
- $G = 2$, ✓ $\quad F = 2$ ✗
- $F = G$ ✗ typing

  $(F=1) = (G=1)$ ✗

  $G = \underbrace{(F=1)}_{\text{st. ass}}$ ✗
- $F = 1 \wedge G = 1$ ✓ ⑤
- $((\neg(F = 1)) \wedge (G = 1))$ ✓
- $\neg(F = 1 \wedge G = 1)$, ✓ $\quad \cancel{(\neg F) = 1 \wedge G = 1}$, $\quad (\neg(F = 1)) \wedge G = 1$ ✓

State Assertions: Semantics

- The **semantics** of **state assertion** P is a function

$$\mathcal{I}[\![P]\!] : \mathsf{Time} \to \{0, 1\},$$

i.e., $\mathcal{I}[\![P]\!](t)$ denotes the truth value of P at time $t \in \mathsf{Time}$.

- The value $\mathcal{I}[\![P]\!](t)$ is defined **inductively** over the structure of P:

$$\mathcal{I}[\![0]\!](t) = 0$$

$$\mathcal{I}[\![1]\!](t) = 1$$

base cases

$$\mathcal{I}[\![X = d]\!](t) = \begin{cases} 1, & \text{if } X_{\mathcal{I}}(t) = \hat{d} \\ 0, & \text{otherwise} \end{cases}$$

induction steps

$$\mathcal{I}[\![\neg P_1]\!](t) = 1 - \mathcal{I}[\![P_1]\!](t)$$

$$\mathcal{I}[\![P_1 \wedge P_2]\!](t) = \begin{cases} 1, & \text{if } \mathcal{I}[\![P_i]\!](t) = 1,\ i \in \{1,2\} \\ 0, & \text{otherwise} \end{cases}$$

State Assertions: Notes

- If X is a boolean observer. the following equalities hold:

$$\underbrace{\mathcal{I}[\![X]\!](t)}_{\text{above}} = \mathcal{I}[\![X = \underbrace{1}_{\substack{\text{boolean}\\ \text{values } (0,1)}}]\!](t) = \mathcal{I}(X)(t) = \underbrace{X_\mathcal{I}(t)}_{\text{abbrev.}}.$$

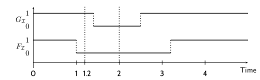
- $\mathcal{I}[\![P]\!]$ is also called **interpretation** of P.

  We shall write $P_\mathcal{I}$ as a **shorthand notation**.

- Here, the state assertions 0 and 1 are treated like boolean values (like tt and ff), it will become clear in a minute, why 0, 1 is a better choice than tt and ff.

State Assertions: Example

- Interpretation $\mathcal{I}$ of **boolean observables** G and F:



$\mathcal{I}[\![L]\!](1.2)$

- Consider **state assertion** $L := \underbrace{G \wedge \neg F}$. $(\,G=1) \wedge \neg\,(F=1))$
- $L_\mathcal{I}(1.2) = 1$, because

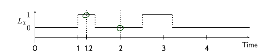
  $\mathcal{I}[\![G \wedge \neg F]\!](t) = 1$ because $\mathcal{I}[\![G]\!](t) = \mathcal{I}[\![G=1]\!](t) = G_\mathcal{I}(t) = 1 = 1$

  $\mathcal{I}[\![\neg F]\!](t) = 1 - \mathcal{I}[\![F=1]\!](t) = 1$

  $\mathcal{I}[\![F=1]\!](t) = (F_\mathcal{I}(t) = 1) = 0$

- $L_\mathcal{I}(2) = 0$, because

- Interpretation of L as timing diagram:



Duration Calculus: Terms

Duration Calculus: Overview

We will introduce **four syntactical categories** (and **abbreviations**):

(i) **Symbols:**

$$\overbrace{true, false, =, <, >, \leq, \geq}^{p,q}, \quad f, g, \quad X, Y, Z, \quad d, \quad x, y, z,$$

(ii) **State Assertions:**

$$P ::= 0 \mid 1 \mid X = d \mid \neg P_1 \mid P_1 \wedge P_2$$

(iii) **Terms:**

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

(iv) **Formulae:**

$$F ::= p(\theta_1, \ldots, \theta_n) \mid \neg F_1 \mid F_1 \wedge F_2 \mid \forall x \bullet F_1 \mid F_1 \,;\, F_2$$

(v) **Abbreviations:**

$$\lceil\rceil, \quad \lceil P \rceil, \quad \lceil P \rceil^t, \quad \lceil P \rceil^{\leq t}, \quad \Diamond F, \quad \Box F$$

Terms: Syntax

- **Duration terms** (or DC terms, or just terms) are defined by the following grammar:

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

  where

  - x is a **global variable** from GVar,
  - f a **function symbol** (of arity n).
  - P is a **state assertion**, and
- 'ℓ' and '$\int$' are like **keywords** (or terminal symbols) in programming languages.
  - ℓ is called **length operator**,
  - $\int$ is called **integral operator**.
- **Notation**: we may write function symbols in **infix notation** as usual, i.e. we may write $\theta_1 + \theta_2$ instead of $\underbrace{+(\theta_1, \theta_2)}_{\text{prefix version form}}$.

Terms: Syntax

- **Duration terms** (or DC terms, or just terms) are defined by the following grammar:

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

  where

  - x is a **global variable** from GVar,
  - f a **function symbol** (of arity n).
  - P is a **state assertion**, and
- 'ℓ' and '$\int$' are like **keywords** (or terminal symbols) in programming languages.
  - ℓ is called **length operator**,
  - $\int$ is called **integral operator**.
- **Notation**: we may write function symbols in **infix notation** as usual, i.e. we may write $\theta_1 + \theta_2$ instead of $+(\theta_1, \theta_2)$.

Definition 1. [*Rigid*]
A term **without** length and integral operators is called **rigid**.

Towards Semantics of Terms: Intervals

- Let $b, e \in$ Time be points in time s.t. $b \le e$. (begin, end)

  Then $[b, e]$ denotes the **closed interval** $\{x \in \text{Time} \mid b \le x \le e\}$.

- We use 'Intv' to denote the set of **closed intervals** in the time domain, i.e.

$$\text{Intv} := \{[b, e] \mid b, e \in \text{Time}\}.$$

- **Closed intervals** of the form $[b, b]$ are called **point intervals**.

Terms: Semantics

- The **semantics** of a **term** θ is a function

$$\mathcal{I}[\![\theta]\!] : \text{Val} \times \text{Intv} \to \mathbb{R},$$

that is, $\mathcal{I}[\![\theta]\!]$ maps a pair consisting of a **valuation** and an **interval** to a real number.

- $\mathcal{I}[\![\theta]\!](\mathcal{V}, [b, e])$ is called
  - the **value** (or **interpretation**) of θ
    - **under interpretation** $\mathcal{I}$ and **valuation** $\mathcal{V}$
      - **in the interval** $[b, e]$.

- The value $\mathcal{I}[\![\theta]\!](\mathcal{V}, [b, e])$ is defined **inductively** over the structure of θ:

base case

$$\mathcal{I}[\![x]\!](\mathcal{V}, [b, e]) = \mathcal{V}(x)$$

$$\mathcal{I}[\![\ell]\!](\mathcal{V}, [b, e]) = e - b$$

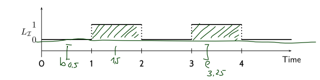
induct. steps

$$\mathcal{I}[\![\textstyle\int P]\!](\mathcal{V}, [b, e]) = \int_b^e P_{\mathcal{I}}(t)\ dt$$ Riemann integral

$$\mathcal{I}[\![f(\theta_1, \dots, \theta_n)]\!](\mathcal{V}, [b, e]) = \hat{f}(\ \mathcal{I}[\![\theta_1]\!](\mathcal{V}, [b,e]), \dots, \mathcal{I}[\![\theta_n]\!](\mathcal{V}, [b,e]))$$

Terms: Example

$$\mathcal{V}(x) = 20.$$



Consider the **term** $\theta = x \cdot \int L$.

- $\mathcal{I}[\![\theta]\!](\mathcal{V}, [0.5, 3.25]) = \mathcal{I}[\![\cdot(x, \int L)]\!](\mathcal{V}, [0.5, 3.25])$

$$= \hat{\cdot}(\quad \mathcal{I}[\![x]\!](\mathcal{V}, [0.5, 3.25]), \quad \mathcal{I}[\![\textstyle\int L]\!](\mathcal{V}, [0.5, 3.25]) \quad)$$

$$= \hat{\cdot}(\quad \mathcal{V}(x), \quad \mathcal{I}[\![\textstyle\int L]\!](\mathcal{V}, [0.5, 3.25]) \quad)$$

$$= \hat{\cdot}(\quad 20, \quad \mathcal{I}[\![\textstyle\int L]\!](\mathcal{V}, [0.5, 3.25]) \quad)$$

$$= \hat{\cdot}\left(\quad 20, \quad \int_{0.5}^{3.25} L_{\mathcal{I}}(t)dt \quad \right) = \hat{\cdot}(\quad 20, \quad 1.25 \quad) = 20 \cdot 1.25 = 25$$

- $\mathcal{I}[\![\theta]\!](\mathcal{V}, [1.5, 1.5]) = 0$

Terms: Is the Semantics Well-defined?

- So, $\mathcal{I}[\![\int P]\!](\mathcal{V}, [b, e])$ is $\displaystyle\int_b^e P_{\mathcal{I}}(t)\ dt$ – but **does the integral always exist**?

- IOW: is there a $P_{\mathcal{I}}$ which is **not (Riemann-)integrable**? Yes. For instance

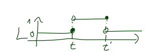
$$P_{\mathcal{I}}(t) = \begin{cases} 1 & \text{, if } t \in \mathbb{Q} \\ 0 & \text{, if } t \notin \mathbb{Q} \end{cases}$$

- To exclude such functions, DC considers only interpretations $\mathcal{I}$ satisfying the following condition of **finite variability**:

  For each state variable X and each interval $[b, e]$ there is a **finite partition** of $[b, e]$ such that the interpretation $X_{\mathcal{I}}$ is **constant on each part**.

Thus a function $X_{\mathcal{I}}$ is of **finite variability** if and only if, on each interval $[b, e]$, the function $X_{\mathcal{I}}$ has only **finitely many points of discontinuity**.

Terms: Remarks



Remark 2.5. The semantics $\mathcal{I}[\![\theta]\!]$ of a term is insensitive against changes of the interpretation $\mathcal{I}$ at individual time points.

More formally:

- Let $\mathcal{I}_1, \mathcal{I}_2$ be interpretations of Obs such that $\mathcal{I}_1(X)(t) = \mathcal{I}_2(X)(t)$ for all $X \in$ Obs and all $t \in \text{Time} \setminus \{t_0, \dots, t_n\}$.
  Then $\mathcal{I}_1[\![\theta]\!](\mathcal{V}, [b, e]) = \mathcal{I}_2[\![\theta]\!](\mathcal{V}, [b, e])$ for all terms θ and intervals $[b, e]$.

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

Syntax / Semantics Overview

Syntax	Semantics (meaning)
predicate symbols	
$true, false, =, <, >, \le, \ge$	$\hat{true} = \text{tt} \in \mathbb{B}, \quad \hat{=} : \mathbb{R}^2 \to \mathbb{B}$
function symbols $f/n, g$	$\hat{f} : \mathbb{R}^n \to \mathbb{R}$
state variables X, Y, Z	$\mathcal{I}(X) : \text{Time} \to \mathcal{D}(X)$
domain values d	$\hat{d} \in \mathcal{D}(X)$
global variables x, y, z	$\mathcal{V}(x) \in \mathbb{R}$
state assertions P	$\mathcal{I}[\![P]\!] : \text{Time} \to \{0, 1\}$
	$\mathcal{I}[\![P]\!](t) \in \{0, 1\}$
terms θ	$\mathcal{I}[\![\theta]\!] : \text{Val} \times \text{Intv} \to \mathbb{R}$
	$\mathcal{I}[\![\theta]\!](\mathcal{V}, [b, e]) \in \mathbb{R}$
formula F	$\mathcal{I}[\![F]\!] : \text{Val} \times \text{Intv} \to \{tt, ff\}$?

Content

- **Symbols**
  - **predicate** and **function symbols**
  - **state variables** and **domain values**
  - **global** (or logical) **variables**
- **State Assertions**
  - **syntax**
  - **semantics**
- **Terms**
  - **syntax**
  - **rigid terms**
  - **intervals**
  - **semantics**
  - **remarks**

Tell Them What You've Told Them…

- **State assertions** over
  - **state variables** (or **observables**), and
  - **predicate symbols**

  are **evaluated** at **points in time**.

  The **semantics** of a **state assertion** is a **truth value**.
- **Terms** are **evaluated** over **intervals** and can
  - measure the **accumulated duration** of a **state assertion**,
  - measure the **length** of intervals, and
  - use **function symbols**.

  The **semantics** of a **term** is a **real number**.
- The value of **rigid terms** is independent from the considered interval.
- The semantics of **terms** is **insensitive** against changes at finitely many **points in time**.

References

References

Olderog, E.-R. and Dierks, H. (2008). *Real-Time Systems - Formal Specification and Automatic Verification.* Cambridge University Press.