TIMED BEHAVIOURAL EQUIVALENCES
How does time affect behavioural equivalences?

Notice: timed behavioural equivalences are valid for

- **Timed Automata**
  as well as
- **Processes expressed in TCCS**

⇒ the equivalences are build on **TLTS**
Content

- Timed and Untimed Trace Equivalence
- Timed and Untimed Bisimilarity
- Weak Timed Bisimilarity
- Region Graphs
Timed and untimed trace equivalence

timed trace, timed language (equivalence),
untimed trace, untimed language (equivalence)
Timed Traces of TLTS

Let $A = (L, l_0, E, I)$ be a timed automaton over a set of clocks $C$ and a set of actions $\text{Act}$.

**Definition: Timed Trace**

A sequence $(t_1, a_1)(t_2, a_2)(t_3, a_3)\ldots$, where $t_i \in \mathbb{R}_{\geq 0}$ and $a_i \in \text{Act}$, is called a finite or infinite timed trace of $A$ iff there is a finite or infinite transition sequence

$$
(l_0, v_0) \xrightarrow{d_1} (l_0, v_1) \xrightarrow{a_1} (l_1, v_2) \xrightarrow{d_2} (l_1, v_3) \xrightarrow{a_2} (l_2, v_4) \ldots
$$

in $T(A)$ such that $v_0(x) = 0$ for all $x \in C$, and, for each $i$, $t_i = t_{i-1} + d_i$ where $t_0 = 0$

**Note that:**

- $t_i$ represents the absolut time stamp at which $a_i$ happen
- The sequence of time stamps is non-decreasing
The following sequence is a timed trace of the above shown automaton:

\[(2.3, \text{press})(2.5, \text{press})(2.51, \text{press})(5.6, \text{press})(5.6, \text{press})(7.0, \text{press})\]

Any non-decreasing sequence of time-stamps induces a timed trace of this timed automaton.
Timed Language

Definition: Timed Language
The set of all finite timed traces of a timed automaton $A$ is denoted by $L(A)$ and is called the timed language of $A$. Timed automata $A_1$ and $A_2$ are timed-language equivalent iff $L(A_1) = L(A_2)$.

Example for two timed-language-equivalent timed-automata

(a) \[ \begin{align*} x \leq 1 & \quad \text{and} \quad x := 0 \quad \text{on} \quad a \\ & \quad \text{on} \quad a \end{align*} \]

(b) \[ \begin{align*} x \leq 1 & \quad \text{and} \quad x := 0 \quad \text{on} \quad a \\ & \quad \text{on} \quad a \]
Untimed Traces + Languages

- In some cases, abstracting from the particular time points and consider only the action sequences might be useful!

**Definition: Untimed Trace**

We say that $a_1a_2a_3\ldots$ is an untyped trace of $A$ iff there exist $t_1, t_2, t_3, \ldots \in \mathbb{R}_{\geq 0}$ such that $(t_1, a_1)(t_2, a_2)(t_3, a_3)\ldots$ is a timed trace of $A$.

**Definition: Untimed Language**

The set of all finite timed traces of a timed automaton $A$ is denoted by $L_u(A)$ and is called the untimed language of $A$. Timed automata $A_1$ and $A_2$ are untimed-language equivalent iff $L_u(A_1) = L_u(A_2)$. 

The converse of the above theorem does not hold, as demonstrated by the following example:

- Example trace: \((0, a)\)
Timed and Untimed Bisimilarity

timed bisimulation, untimed bisimulation
Timed Bisimulation

- Language equivalence is not always the most suitable notion of behavioural equivalence to consider.
- Strong bisimilarity can be generalized to timed bisimilarity.

**Definition: Timed Bisimulation**

A binary relation $R$ over the set of states of a TLTS is a *timed bisimulation* iff whenever $s_1 R s_2$, $a$ is an action and $d$ is a time delay:

if $s_1 \xrightarrow{a} s_1'$ then there is a transition $s_2 \xrightarrow{a} s_2'$ such that $s_1' R s_2'$. 

Two states $s$ and $s'$ are timed bisimilar, written $s \sim s'$, iff there is a timed bisimulation that relates them.
Example for Timed Bisimulation

Definition: Timed Bisimulation for Timed-Automata

Timed automata $A_1$ and $A_2$ are timed bisimilar iff their initial states in the union of the timed transition systems $T(A_1)$ and $T(A_2)$ generated by $A_1$ and $A_2$ are timed bisimilar.

- The strong bisimulation relation $R$, where

  $((A, v_0), (A', v_0)) \in R$

  holds:

  $$\{((A, [x = d]), (A', [x = d])) \mid d \in \mathbb{R}_{\geq 0}\}$$

  $\cup \{((B, [x = d + 1]), (B', [x = d])) \mid d \in \mathbb{R}_{\geq 0}\}$

  $\cup \{((C, [x = d]), (C', [x = d'])) \mid d, d' \in \mathbb{R}_{\geq 0}\}.$
Untimed Bisimulation

- Again, we want to abstract from time.

**Definition: Untimed Bisimulation**

A binary relation $R$ over the set of states of a TLTS is a *untimed bisimulation* iff whenever $s_1 R s_2$, $a$ is an action and $d$ is a time delay:

- If $s_1 \xrightarrow{a} s'_1$ then there is a transition $s_2 \xrightarrow{a} s'_2$ such that $s_1 R s'_2$.

- If $s_2 \xrightarrow{a} s'_2$ then there is a transition $s_1 \xrightarrow{a} s'_1$ such that $s'_1 R s'_2$.

Two states $s$ and $s'$ are untimed bisimilar, written $s \sim s'$, iff there is a untimed bisimulation that relates them.
Example for Untimed Bisimulation

These timed non-bisimilar automata are equivalent with respect to untimed bisimilarity.

Consider the following relation $R$:

\[
\{ ((A, [x = d]), (A', [x = d'])) | 0 \leq d \leq 1, 0 \leq d' \leq 2 \} \\
\cup \{ ((A, [x = d]), (A', [x = d'])) | d > 1, d' > 2 \} \\
\cup \{ ((B, [x = d]), (B', [x = d'])) | d, d' \in \mathbb{R}_{\geq 0} \}. 
\]
Weak timed bisimilarity

weak timed bisimulation, weakly timed bisimilar
Abstracting from $\tau$-actions

- ...in a timed setting. What does that mean?

Consider these example TCCS Processes:

1. $\tau.a.\tau.P$
2. $a.P$

a) $\varepsilon(4).\tau.\varepsilon(3).P$

b) $\varepsilon(4).\varepsilon(3).P$

c) $\varepsilon(7).P$
Extended Notion of TLTS

**Definition: ETLTS**

Let $T = \left(\text{Proc, Lab, } \left\{ \frac{\alpha}{\rightarrow} \mid \alpha \in \text{Lab} \right\} \right)$ be a timed transition system where $\text{Lab} = \text{Act} \cup \mathbb{R}_{\geq 0}$ and let $s, t$ be states of $T$. For each action $\alpha \in \text{Lab}$, we shall write $s \xrightarrow{\alpha} t$ iff

- $\alpha = \tau$ and $s(\frac{\tau}{\rightarrow})^* t$, or
- $\alpha = a \in \text{Act}\backslash \{\tau\}$ and $s \xrightarrow{\tau} s_1 \xrightarrow{a} s_2 \xrightarrow{\tau} t$ for some states $s_1$ and $s_2$, or
- $\alpha = d \in \mathbb{R}_{\geq 0}$ and $s \xrightarrow{\tau} s_1 \xrightarrow{d_1} t_1 \cdots t_{n-1} \xrightarrow{\tau} s_n \xrightarrow{d_n} t_n \xrightarrow{\tau} t$ for some $n \geq 0$, states $s_1, \ldots, s_n, t_1, \ldots, t_n$ and delays $d_1, \ldots, d_n$ with $d = \sum_{i=1}^{n} d_i$. By convention, $d = 0$ when $n = 0$.

- Consider the TCCS process $\varepsilon(4) \cdot \tau \cdot \varepsilon(3) \cdot P$
- The process affords the transition $\varepsilon(4) \cdot \tau \cdot \varepsilon(3) \cdot P \xrightarrow{\tau} P$
Weak Timed Bismulation

**Definition: Weak Timed Bisimulation**

A binary relation $R$ over the set of states of a TLTS is a weak timed bisimulation iff whenever $s_1 R s_2$, $a$ is an action (including $\tau$) and $d$ is a time delay:

- if $s_1 \xrightarrow{a} s'_1$ then there is a transition $s_2 \xrightarrow{a} s'_2$ such that $s'_1 R s'_2$
- if $s_1 \xrightarrow{d} s'_1$ then there is a transition $s_2 \xrightarrow{d} s'_2$ such that $s'_1 R s'_2$
- if $s_2 \xrightarrow{a} s'_2$ then there is a transition $s_1 \xrightarrow{a} s'_1$ such that $s'_1 R s'_2$
- if $s_2 \xrightarrow{d} s'_2$ then there is a transition $s_1 \xrightarrow{d} s'_1$ such that $s'_1 R s'_2$

Two states $s$ and $s'$ are weakly timed bisimilar, written $s \approx s'$, iff there is a weak timed bisimulation that relates them.

**Definition: Weak Timed Bisimulation for Timed-Automata**

Timed automata $A_1$ and $A_2$ are weakly timed bisimilar iff their initial states in the union of the timed transition systems $T(A_1)$ and $T(A_2)$ generated by $A_1$ and $A_2$ are weakly timed bisimilar.
Example for Weak Timed Bisimulation

Consider the following two TCCS processes $P$ and $Q$

$P = (a.\varepsilon(3).b.0 \mid b.\varepsilon(4).c.0)\backslash b$

$Q = a.\varepsilon(7).c.0$

Are $P$ and $Q$ weakly timed bisimilar?

Yes, indeed!

A weak timed bisimulation $R$ containing $(P, Q)$:

$\{(\varepsilon(3-d).b.0 \mid b.\varepsilon(4).c.0)\backslash b, \varepsilon(7-d).c.0) \mid 0 \leq d \leq 3\}$

$\cup \{((0 \mid \varepsilon(4-d).c.0)\backslash b, \varepsilon(4-d).c.0) \mid 0 \leq d \leq 4\}$

$\cup \{(P, Q)\}$.
Content

- Timed and Untimed Trace Equivalence
- Timed and Untimed Bisimilarity
- Weak Timed Bisimilarity
- Region Graphs
Support for Algorithmic Analysis

- Due to valuation clocks, even simple timed automata generate TTS infinitely many reachable states.

Is algorithmic verification at all possible over timed automata?

- Yes! Idea: partitioning clock valuations into finitely many equivalence classes.
  - Any two clock valuations from the same equivalence class will not create any ‘significant difference’.

  Region-Graph Technique by Alur and Dill (1990, 1992, 1994)
Formal goal: equivalence relation \(\equiv\) over clock valuations

where \(\equiv\subseteq (C \rightarrow \mathbb{R}_{\geq 0}) \times (C \rightarrow \mathbb{R}_{\geq 0})\) is such that

1. \(v \equiv v'\) implies that the states \((l, v)\) and \((l, v')\) are untimed bisimilar for each location of the automaton \(A\), and

2. \(\equiv\) has only finitely many equivalence classes, i.e. the set \([v]_{\equiv} = \{v' | v' \equiv v\}\) is finite. We shall call \([v]_{\equiv}\) the equivalence class represented by \(v\).
Idea for the Nature of Region Graphs

\[ s_1 = (\ell_0, [x = 3, y = 0.25]) \]

\[ s_2 = (\ell_0, [x = 3, y = 1.75]) \]
How many regions are there overall?
- 6 corner points,
- 8 closed line segments,
- 4 open line segments,
- 3 closed areas and
- 3 open areas

24 regions!

Proposition

Let $A$ be a timed automaton with $n$ clocks. Let $C$ be the set of clocks in $A$. The number of regions of $A$ is bounded by

$$2^n \cdot n! \cdot \prod_{x \in C} (2c_x + 2).$$

- The number of regions is exponential in the number of clocks and,
- in the constants mentioned in the clock constraints.

Alur and Dill (1994)
What Problems can be tackled with Region Graphs?

- Reachability and untimed-language equivalence for timed automata is decidable in PSPACE (polynomial space) (Alur and Dill, 1994).

- Untimed bisimilarity for timed automata has been proved decidable in EXPTIME (deterministic exponential time) (Larsen and Yi, 1997).

- Timed bisimilarity was shown to be decidable in EXPTIME Cerans (1993).

- Timed-language equivalence for timed automata is known to be undecidable (Alur and Dill, 1994).
Zones

- More efficient representation of the state space for timed automata
- Zones are unions of regions:
  - coarser
  - more compact

\[(\ell_0, x = y = 0) \leadsto (\ell_0, x = y)\]
\[\leadsto (\ell_0, y = 0 \land x \leq 2)\]
\[\leadsto (\ell_0, y \leq x \leq y + 2)\]
\[\leadsto (\ell_0, y = 0 \land x \leq 4)\]
\[\leadsto (\ell_0, y \leq x \leq y + 4)\]
\[\leadsto (\ell_1, y \leq 2 \land 4 \leq x \leq y + 4)\]

Example TA
Is \(l_1\) reachable?

Symbolic States

Zones

Bengtsson and Yi (2003)
Summary
Timed and Untimed Trace Equivalence
- Still a very „strict“ equivalence relation, as seen in chapter 3

Timed and Untimed Bisimilarity
- More relaxed equivalence relation

Weak Timed Bisimilarity
- Further relaxed equivalence relation

Region Graphs
- Finite abstraction of infinitite transition systems
- Zones introduce further abstraction