Compositionality in Real-Time Model Checking

Jasper Berendsen & Frits Vaandrager

Radboud University Nijmegen

Foundations of Interface Technology (FIT) Workshop
April 5, 2008

A Dichotomy

Modeling languages for reactive systems typically either support communication via shared variables or communication via synchronization of actions:

- TLA, Reactive Modules, etc,
- CCS, I/O automata, ACP, mCRL2, etc

A Non-Issue?

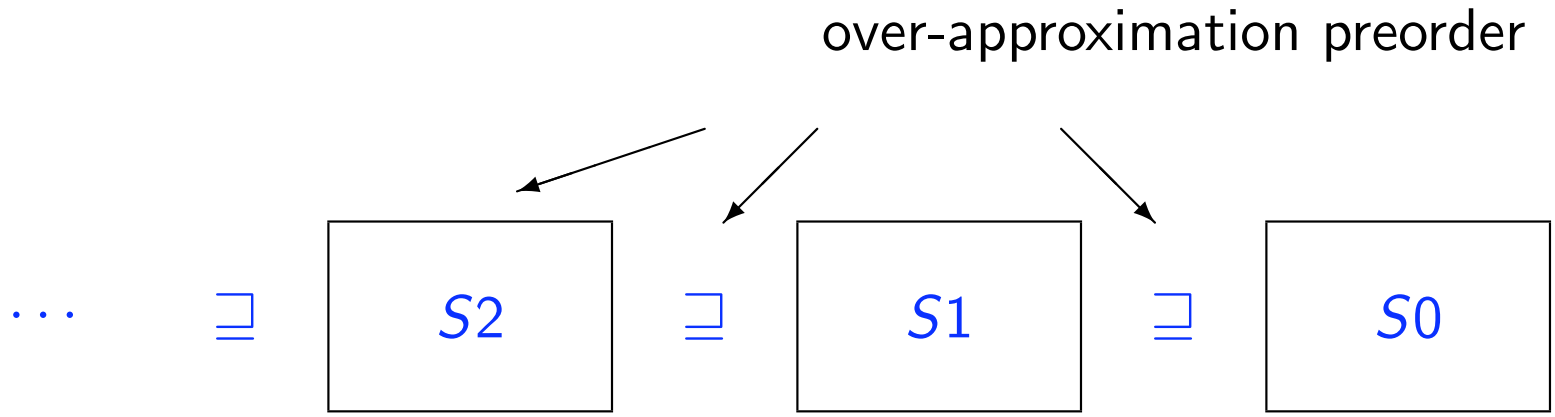
Both types of communication can be defined in terms of each other:

- A shared variable can be modeled as a separate process/automaton that communicates with its environment via read/write synchronization actions.
- Synchronization of actions can be modeled using some auxiliary flag variables and handshake transitions of the synchronizing automata.

However, these encodings blow up the state space and make it more difficult to understand the model!
We want to have it all!

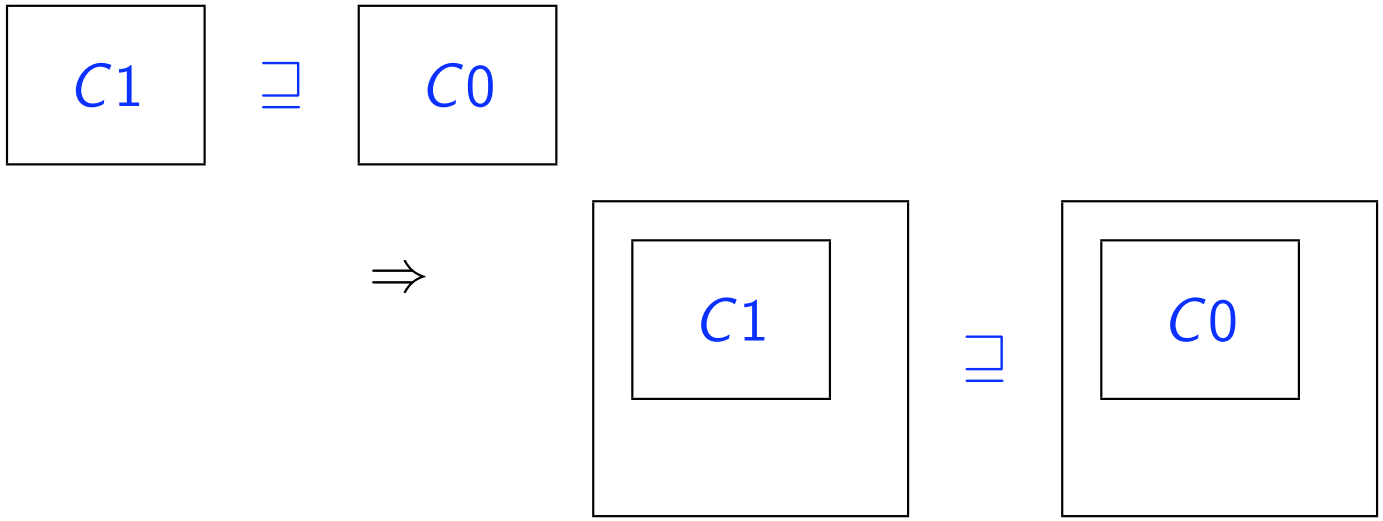
Refinement

Abstraction/refinement is a key technique to combat state space explosions in model checking.



Compositionality

Compositional abstraction is even more useful!



Our Result

A framework for compositional abstraction for UPPAAL based on simulation relations that does support synchronization of actions, communication via shared variables, and committed locations.

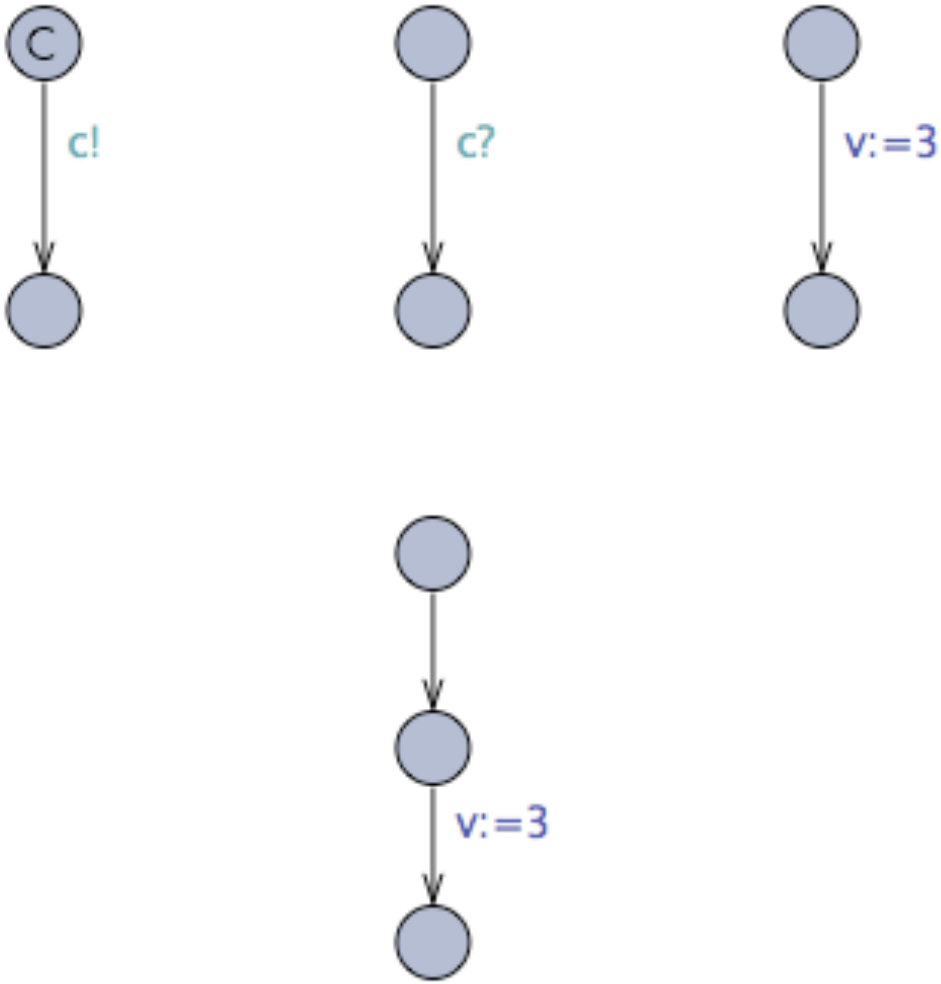
Uppaal

Model checker for timed automata developed originally by universities of Uppsala and Aalborg, with recent contributions by Nijmegen. Many industrial applications!

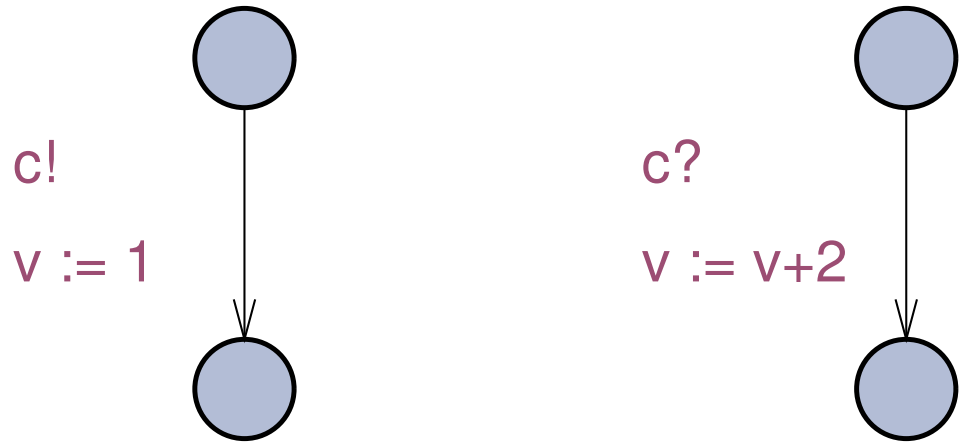
- Bang & Olufsen protocol, Philips Audio Control
- Biphase Mark protocol
- IEEE 1394 "Firewire", Zeroconf, SHIM6
- scheduling of lacquer production at Axxom
- throughput optimization for a wafer scanner from ASML
- car periphery supervision system from Bosch
- architecture evaluation for a distributed in-car navigation system by Siemens
- mutex and semaphore examples
- ...

Uppaal Modelling Language

- Networks of Timed Automata
- Uppaal supports both shared variables and synchronization of actions *CCS style*
- Recently, Uppaal has been extended with C-like functions and the verification engine has become much more powerful (e.g. due to symmetry reduction).
- Many other features: committed locations, urgent channels, broadcast communication,..


C
c!
c?
v:=3
v:=3

Combining the Two Means for Communication



Proposals:
(a) Rule out syntactically: only one component has write permission for each variable;
(b) Rule out semantically: results of both assignments should be the same;
(c) First perform assignment for $c!$, then assignment for $c?$.

Actions

We consider labeled transition systems with 3 types of state transitions, corresponding to 3 distinct sets of actions:

1. The set of *external actions* is defined as $\mathcal{E} \triangleq \{c!, c? \mid c \in \mathcal{C}\}$, where $\mathcal{C}$ denotes the set of *channels*.
2. We assume the existence of a special *internal action* τ.
3. Finally, we assume a set of *durations* or *time-passage actions*, which are just the nonnegative real numbers in $\mathbb{R}_{\geq 0}$.

Semantics for real-time systems

A *timed transition system (TTS)* is a tuple

$$\mathcal{T} = \langle E, H, S, s^0, \longrightarrow^1, \longrightarrow^0 \rangle,$$

where $E, H \subseteq \mathcal{V}$ are disjoint sets of external and internal variables, respectively, $V = E \cup H$, $S \subseteq Val(V)$, and $\langle S, s^0, Act, \longrightarrow^1 \cup \longrightarrow^0 \rangle$ is an LTS.
We write $r \xrightarrow{a,b} s$ if $(r, a, s) \in \longrightarrow^b$.

Semantics for real-time systems

A *timed transition system (TTS)* is a tuple

$$\mathcal{T} = \langle E, H, S, s^0, \longrightarrow^1, \longrightarrow^0 \rangle,$$

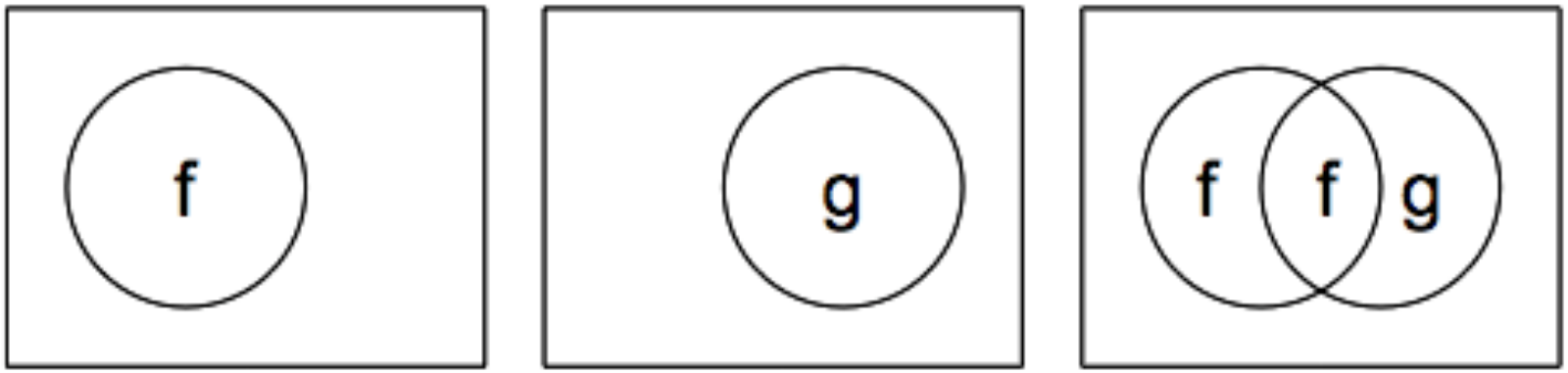
where $E, H \subseteq \mathcal{V}$ are disjoint sets of external and internal variables, respectively, $V = E \cup H$, $S \subseteq Val(V)$, and $\langle S, s^0, Act, \longrightarrow^1 \cup \longrightarrow^0 \rangle$ is an LTS.
We write $r \xrightarrow{a,b} s$ if $(r, a, s) \in \longrightarrow^b$.

But for a compositional framework some axioms are *needed* on TTSs.

Notation

For functions f and g, the *left-merge*, written $f \rhd g$ denotes the function defined on both domains, where:
f overrides g for all elements in the intersection of their domains:

$$(f \rhd g)(z) \triangleq \begin{cases} f(z) & \text{if } z \in dom(f) \\ g(z) & \text{if } z \in dom(g) - dom(f) \end{cases}$$



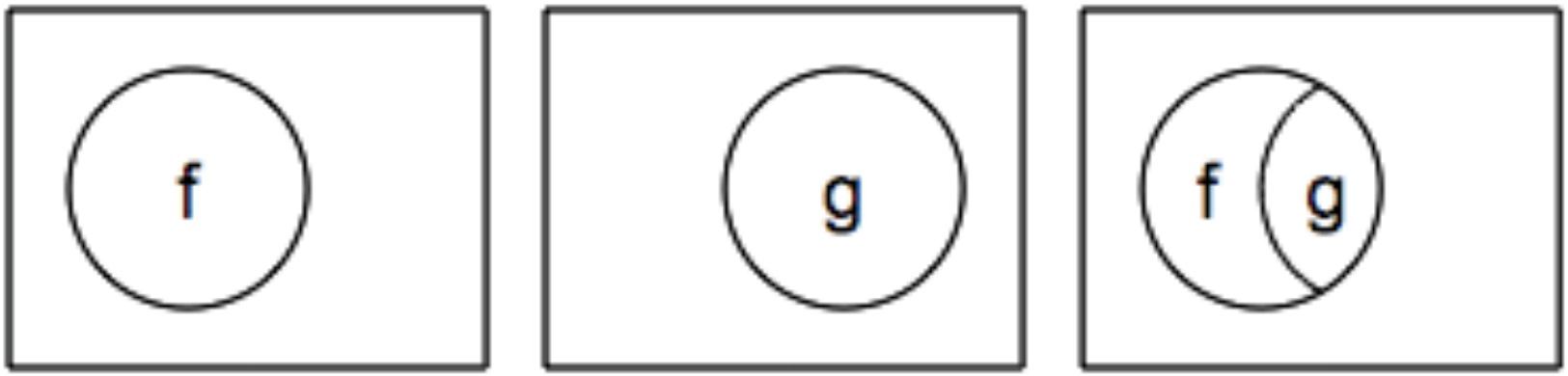
right-merge definition $f \lhd g \triangleq g \rhd f$.

Notation (continued)

Two functions f and g are *compatible* if $f \rhd g = f \lhd g$.

For compatible functions f and g, *merge* is defined as $f \| g \triangleq f \rhd g$.

We write $f[g]$ for the *update* of function f according to g, that is, $f[g] \triangleq (f \lhd g) \lceil dom(f)$.



Laws

$$(f \rhd g) \rhd h = f \rhd (g \rhd h)$$

$$f[g][h] = f[h \rhd g]$$

$$f \| (g \| h) = (f \| g) \| h$$

$$f \| g = g \| f$$

$$f \rhd g = f \| g[f]$$

$$(f \rhd g)[h] = f[h] \rhd g[h]$$

Axioms for TTS

We require the following axioms, for all $s, t \in S$, $a, a' \in Act$, $b \in \mathbb{B}$, $d \in \mathbb{R}_{\geq 0}$ and $u \in Val(E)$,

$$Comm(s) \wedge s \xrightarrow{a',b} \Rightarrow \quad a' \in \mathcal{E} \vee (a' = \tau \wedge b = 1)$$

$$s[u] \in S$$

$$s \xrightarrow{c?,b} \Rightarrow \quad s[u] \xrightarrow{c?,b}$$

$$s \xrightarrow{d,0} t \quad \Rightarrow \quad t \text{ where clocks incremented by } d$$

A state s of a TTS is called committed, notation $Comm(s)$, iff it enables an outgoing committed transition.

Rules for Parallel Composition of TTSs

Composition $\mathcal{T}_1 \| \mathcal{T}_2$, and $i, j \in 1, 2$

$$\frac{r \xrightarrow{e,b}_i r'}{r \| s \xrightarrow{e,b} r' \rhd s}$$

Rules for Parallel Composition of TTSs

Composition $\mathcal{T}_1 \| \mathcal{T}_2$, and $i, j \in 1, 2$

$$\frac{r \xrightarrow{e,b}_i r'}{r \| s \xrightarrow{e,b} r' \rhd s}$$

$$\frac{r \xrightarrow{\tau,b}_i r' \qquad Comm(s) \Rightarrow b}{r \| s \xrightarrow{\tau,b} r' \rhd s}$$

Rules for Parallel Composition of TTSs

Composition $\mathcal{T}_1 \| \mathcal{T}_2$, and $i, j \in 1, 2$

$$\frac{r \xrightarrow{e,b}_i r'}{r \| s \xrightarrow{e,b} r' \rhd s}$$

$$\frac{r \xrightarrow{c!,b}_i r' \qquad s[r'] \xrightarrow{c?,b'}_j s' \qquad i \neq j \qquad Comm(r) \vee Comm(s) \Rightarrow b \vee b'}{r \| s \xrightarrow{\tau, b \vee b'} r' \lhd s'}$$

$$\frac{r \xrightarrow{\tau,b}_i r' \qquad Comm(s) \Rightarrow b}{r \| s \xrightarrow{\tau,b} r' \rhd s}$$

Rules for Parallel Composition of TTSs

Composition $\mathcal{T}_1 \| \mathcal{T}_2$, and $i, j \in 1, 2$

Recall Ax 3: $s \xrightarrow{c?,b} \Rightarrow \quad s[u] \xrightarrow{c?,b}$

$$\frac{r \xrightarrow{e,b}_i r'}{r \| s \xrightarrow{e,b} r' \rhd s}$$

$$\frac{r \xrightarrow{c!,b}_i r' \qquad s[r'] \xrightarrow{c?,b'}_j s' \qquad i \neq j \qquad Comm(r) \vee Comm(s) \Rightarrow b \vee b'}{r \| s \xrightarrow{\tau, b \vee b'} r' \lhd s'}$$

$$\frac{r \xrightarrow{\tau,b}_i r' \qquad Comm(s) \Rightarrow b}{r \| s \xrightarrow{\tau,b} r' \rhd s}$$

Rules for Parallel Composition of TTSs

Composition $\mathcal{T}_1 \| \mathcal{T}_2$, and $i, j \in 1, 2$

Recall Ax 3: $s \xrightarrow{c?,b} \Rightarrow \quad s[u] \xrightarrow{c?,b}$

$$\frac{r \xrightarrow{e,b}_i r'}{r \| s \xrightarrow{e,b} r' \rhd s}$$

$$\frac{r \xrightarrow{c!,b}_i r' \qquad s[r'] \xrightarrow{c?,b'}_j s' \qquad i \neq j \qquad Comm(r) \vee Comm(s) \Rightarrow b \vee b'}{r \| s \xrightarrow{\tau, b \vee b'} r' \lhd s'}$$

$$\frac{r \xrightarrow{\tau,b}_i r' \qquad Comm(s) \Rightarrow b}{r \| s \xrightarrow{\tau,b} r' \rhd s}$$

$$\frac{r \xrightarrow{d,0}_i r' \qquad s \xrightarrow{d,0}_j s' \qquad i \neq j}{r \| s \xrightarrow{d,0} r' \| s'}$$

Compositionality

Lemma

Let $\mathcal{T}_1$ and $\mathcal{T}_2$ be compatible TTSs. Then $\mathcal{T}_1 \| \mathcal{T}_2$ is a TTS.

Theorem (Commutativity)

Let $\mathcal{T}_1$ and $\mathcal{T}_2$ be compatible TTSs. Then
$\mathcal{T}_1 \| \mathcal{T}_2 = \mathcal{T}_2 \| \mathcal{T}_1$.

Theorem (Associativity)

Let $\mathcal{T}_1$, $\mathcal{T}_2$ and $\mathcal{T}_3$ be pairwise compatible TTSs. Then
$(\mathcal{T}_1 \| \mathcal{T}_2) \| \mathcal{T}_3 = \mathcal{T}_1 \| (\mathcal{T}_2 \| \mathcal{T}_3)$.

Timed Step Simulation

Given TTSs $\mathcal{T}_1$ and $\mathcal{T}_2$, we say that a relation $\mathrm{R} \subseteq S_1 \times S_2$ is a *timed step simulation* from $\mathcal{T}_1$ to $\mathcal{T}_2$, provided that $E_1 = E_2$, $s_1^0 \,\mathrm{R}\, s_2^0$, and if $s \,\mathrm{R}\, r$ then

1. $s\lceil E_1 = r\lceil E_2$,
2. $\forall u \in \mathit{Val}(E_1) : s[u] \;\mathrm{R}\; r[u]$,
3. if $\mathit{Comm}(r)$ then $\mathit{Comm}(s)$,
4. if $s \xrightarrow{a,b} s'$ then either there exists an r' such that $r \xrightarrow{a,b} r'$ and $s' \,\mathrm{R}\, r'$, or $a = \tau$ and $s' \,\mathrm{R}\, r$.

We write $\mathcal{T}_1 \preceq \mathcal{T}_2$ when there exists a timed step simulation from $\mathcal{T}_1$ to $\mathcal{T}_2$.

Compositionality

Theorem

Let $\mathcal{T}_1, \mathcal{T}_2, \mathcal{T}_3$ be TTSs with $\mathcal{T}_1 \preceq \mathcal{T}_2$, and both $\mathcal{T}_1$ and $\mathcal{T}_2$ compatible with $\mathcal{T}_3$. Then $\mathcal{T}_1 \| \mathcal{T}_3 \preceq \mathcal{T}_2 \| \mathcal{T}_3$.

Consistency with Uppaal Semantics

Theorem

Let $\mathcal{N} = \langle \mathcal{A}_1, \ldots, \mathcal{A}_n \rangle$ be a network of timed automata. Then

$$\mathsf{LTS}(\mathcal{N}) \;\cong\; \mathsf{LTS}((\mathsf{TTS}(\mathcal{A}_1) \| \cdots \| \mathsf{TTS}(\mathcal{A}_n)) \backslash \mathcal{C}).$$

Future Work

- Applicable to the whole Uppaal modelling language, so including broadcast synchronization, and urgency.
- However, *"feature interaction"* requires usage of a theorem prover.
- Extensions with prices, probabilities etc.

A Timed Transition System is defined similarly as before, but the set of external variables E is divided in sets:

- readable variables R
- writable variables W
- internally writable variables $IW \subseteq W$, only writable by this component

$$\frac{r \xrightarrow{a}_i r' \qquad s \xrightarrow{\bar{a}}_j s' \qquad i \neq j \qquad r' \lceil (W_i \cap W_j) = s' \lceil (W_i \cap W_j)}{r \| s \xrightarrow{\tau} (\ (r' \lceil W_i) \parallel (s' \lceil W_j)\ ) \ \triangleright\ (r \| s)}$$

Assume three timed transition systems $\mathcal{T}_A$, $\mathcal{T}_B$, and $\mathcal{T}_C$.

TTS $\mathcal{T}_A$ has $W_A = \{u\}$, $H_A = \{a\}$ and only one transition

$$\{a \mapsto 0, u \mapsto 0\} \xrightarrow{a} \{a \mapsto 1, u \mapsto 1\}$$

TTS $\mathcal{T}_B$ has $W_B = \emptyset$, $H_B = \{b\}$ and only one transtion

$$\{b \mapsto 0, u \mapsto 0\} \xrightarrow{\bar{a}} \{b \mapsto 1, u \mapsto 0\}$$

TTS $\mathcal{T}_C$ has $W_C = \{u\}$, $H_C = \{c\}$ and only one state

$$\{c \mapsto 0, u \mapsto 0\}$$

$$\{a \mapsto 0, b \mapsto 0, u \mapsto 0\} \xrightarrow{\tau} \{a \mapsto 1, b \mapsto 1, u \mapsto 1\}$$

$$\{a \mapsto 0, b \mapsto 0, c \mapsto 0, u \mapsto 0\} \xrightarrow{\tau} \{a \mapsto 1, b \mapsto 1, c \mapsto 0, u \mapsto 1\}$$

$$\{b \mapsto 0, c \mapsto 0, u \mapsto 0\} \xrightarrow{\bar{a}} \{b \mapsto 1, c \mapsto 0, u \mapsto 0\}$$