Action Abstraction in Timed Process Algebra

The Case for an Untimed Silent Step

Michel A. Reniers   Muck van Weerdenburg

ProSe
January 11, 2007

Outline

Introduction to Timed Process Algebra
- Abstraction in Timed Process Algebra
- Some typical examples
- Illustration: PAR protocol

Untimed Silent Step
- Some typical examples revisited
- Axiomatisation
- Illustration: PAR protocol revisited

Concluding Remarks

Timed Process Algebra

Time domain Time:

- Time is totally ordered by $\leq$;
- smallest element w.r.t. $\leq$ exists and is denoted 0;

Signature $\mathrm{BSP}^{@}_{\mathrm{abs}}$ (*Basic Sequential Processes* with *absolute time* and *time-stamping*): for each $t \in \mathrm{Time}$ and $a \in \mathrm{Act}$

- $0^{@t}$: timed deadlock constant
- $1^{@t}$: timed termination constant
- $a^{@t}.\_$: action prefix operator
- $\_ + \_$: alternative composition
- $t \gg \_$: time-initialisation operator

Semantics - Structured Operational Semantics

$$\frac{}{0^{@t} \rightsquigarrow_u} \; [u \leq t]$$

$$\frac{}{1^{@t} \downarrow_t} \qquad \frac{}{1^{@t} \rightsquigarrow_u} \; [u \leq t]$$

$$\frac{}{a^{@t}.x \xrightarrow{a}_t t \gg x} \qquad \frac{}{a^{@t}.x \rightsquigarrow_u} \; [u \leq t]$$

- $\_ \xrightarrow{a}_t \_$ representing the execution of an action a at time t
- $\_ \downarrow_t$, representing successful termination at time t
- $\_ \rightsquigarrow_t$, representing that the process can delay until at least time t

Semantics - Structured Operational Semantics (cont'd)

$$\frac{x \xrightarrow{a}_t x'}{\begin{array}{c} x + y \xrightarrow{a}_t x' \\ y + x \xrightarrow{a}_t x' \end{array}} \qquad \frac{x \downarrow_t}{\begin{array}{c} x + y \downarrow_t \\ y + x \downarrow_t \end{array}} \qquad \frac{x \rightsquigarrow_t}{\begin{array}{c} x + y \rightsquigarrow_t \\ y + x \rightsquigarrow_t \end{array}}$$

$$\frac{x \xrightarrow{a}_u x'}{t \gg x \xrightarrow{a}_u x'} \; [t \leq u] \qquad \frac{x \downarrow_u}{t \gg x \downarrow_u} \; [t \leq u]$$

$$\frac{}{t \gg x \rightsquigarrow_u} \; [u \leq t] \qquad \frac{x \rightsquigarrow_u}{t \gg x \rightsquigarrow_u}$$

Semantics - Equivalence

A symmetric binary relation R on closed terms is a *timed strong bisimulation* relation if for all r and s with $(r, s) \in R$, we have:

1. if $r \xrightarrow{a}_t r'$, then there exists a s' such that $s \xrightarrow{a}_t s'$ and $(r', s') \in R$;
2. if $r \downarrow_t$, then $s \downarrow_t$;
3. if $r \leadsto_t$, then $s \leadsto_t$.

Two closed terms are strongly bisimilar, notation $p \underline{\leftrightarrow} q$, iff there is a timed strong bisimulation relation R such that $(p, q) \in R$.

Further remarks

- based on the process theory BSP from (Baeten, 2003; Basten, Baeten & Reniers, 2005)
- timing extensions with absolute-timing and time-stamping (both syntactically and semantically) inspired by the process algebra *timed μCRL* (Groote, 1997; Reniers et al, 2002)
- timed strong bisimilarity is a congruence for all operators from this process algebra
- sound and complete axiomatisation of timed strong bisimilarity

Abstraction in Timed Process Algebra

Extensions to the signature of the process algebra: for each $t \in$ Time and $I \subseteq$ Act

- $\tau^{@t}.\_$: silent step prefix operator
- $\tau_I(\_)$: abstraction operator

Semantics - Structured Operational Semantics

$$\frac{}{\tau^{@t}.x \xrightarrow{\tau}_t t \gg x} \qquad \frac{}{\tau^{@t}.x \rightsquigarrow_u} \; [u \le t]$$

$$\frac{x \xrightarrow{\tau}_t x'}{\begin{array}{c} x + y \xrightarrow{\tau}_t x' \\ y + x \xrightarrow{\tau}_t x' \end{array}} \qquad \frac{x \xrightarrow{\tau}_u x'}{t \gg x \xrightarrow{\tau}_u x'} \; [t \le u]$$

Semantics - Structured Operational Semantics (cont'd)

$$\frac{x \xrightarrow{a}_t x'}{\tau_I(x) \xrightarrow{a}_t \tau_I(x')} [a \notin I] \qquad \frac{x \xrightarrow{a}_t x'}{\tau_I(x) \xrightarrow{\tau}_t \tau_I(x')} [a \in I]$$

$$\frac{x \xrightarrow{\tau}_t x'}{\tau_I(x) \xrightarrow{\tau}_t \tau_I(x')} \qquad \frac{x \downarrow_t}{\tau_I(x) \downarrow_t} \qquad \frac{x \rightsquigarrow_t}{\tau_I(x) \rightsquigarrow_t}$$

Semantics - Equivalence

Klusener: idle branching bisimulation

An action transition a at time t may be mimicked by a well-timed sequence of silent steps that is ultimately followed by an a-transition at time t.

- congruence of timed strong bisimilarity
- sound and complete axiomatisation of timed strong bisimilarity

Some typical examples (I)

$$\tau_{\{b\}}(a^{@1}.b^{@2}.c^{@4}.0^{@5})$$

$$\tau_{\{b\}}(a^{@1}.b^{@3}.c^{@4}.0^{@5})$$

$$a^{@1}.c^{@4}.0^{@5}$$

Klusener: all equal

Some typical examples (II)

$$\tau_{\{b\}}(a^{@1}.(b^{@2}.(c^{@3}.0^{@4} + d^{@3}.0^{@4}) + d^{@3}.0^{@4}))$$

$$a^{@1}.(c^{@3}.0^{@4} + d^{@3}.0^{@4})$$

Klusener: different

Some typical examples (III)

$$\tau_{\{b\}}(a^{@1}.(b^{@2}.c^{@3}.0^{@4} + d^{@3}.0^{@4}))$$

$$\tau_{\{b\}}(a^{@1}.(c^{@3}.0^{@4} + b^{@2}.d^{@3}.0^{@4}))$$

Klusener: equal

Some typical examples (IV)

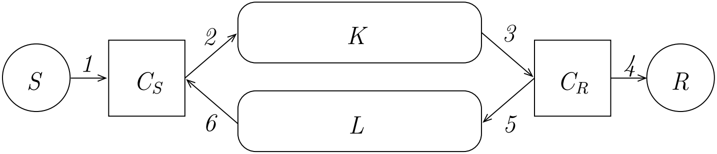
$$\tau_{\{b\}}(a^{@1}.(b^{@3}.0^{@4} + c^{@2}.0^{@4}))$$

$$a^{@1}.(0^{@4} + c^{@2}.0^{@4})$$

Klusener: equal

PAR protocol

Timed version of the Alternating Bit Protocol.



The protocol entities are a sender S, a receiver R, a forward channel K, and a backward channel L.

Channels are unreliable, but errors are observable.

PAR protocol - Timed Silent Step

$$
\begin{aligned}
X_{b,t} &= \sum_{t'} \sum_{d \in D} r_1(d)^{@t+t'} . Y_{d,b,t+t'+t_S} \\
Y_{d,b,t} &= \tau^{@t+t_K} . s_2(d)^{@t+t_K+t_R} . Z_{d,b,t+t_K+t_R+t'_R} \\
&+ \sum_{k \leq t_K} \tau^{@t+k} . Y_{d,b,t+t'_S} \\
Z_{d,b,t} &= \tau^{@t+t_L} . X_{\overline{b},t+t_L} + \sum_{l \leq t_L} \tau^{@t+l} . U_{d,b,t+t'_S-t_K-t_R-t'_R} \\
U_{d,b,t} &= \tau^{@t+t_K} . V_{d,b,t+t_K+t'_R} + \sum_{k \leq t_K} \tau^{@t+k} . U_{d,b,t+t'_S} \\
V_{d,b,t} &= \tau^{@t+t_L} . X_{\overline{b},t+t_L} + \sum_{l \leq t_L} \tau^{@t+l} . U_{d,b,t+t'_S-t_K-t'_R}
\end{aligned}
$$

Outline

Introduction to Timed Process Algebra
- Abstraction in Timed Process Algebra
- Some typical examples
- Illustration: PAR protocol

Untimed Silent Step
- Some typical examples revisited
- Axiomatisation
- Illustration: PAR protocol revisited

Concluding Remarks

Untimed Silent Step

The silent step is not observable nor is its timing

- $\tau.\_$: silent step prefix operator
- $\tau_I(\_)$: abstraction operator

Semantics - Structured Operational semantics

$$\frac{}{\tau.x \xrightarrow{\tau} x}$$

$$\frac{x \xrightarrow{\tau} x'}{\begin{array}{c} x + y \xrightarrow{\tau} x' \\ y + x \xrightarrow{\tau} x' \end{array}} \qquad \frac{x \xrightarrow{\tau} x'}{t \gg x \xrightarrow{\tau} t \gg x'}$$

Semantics - Structured Operational semantics

$$\frac{x \xrightarrow{a}_t x'}{\tau_I(x) \xrightarrow{a}_t \tau_I(x')} \; [a \notin I] \qquad \frac{x \xrightarrow{a}_t x'}{\tau_I(x) \xrightarrow{\tau} \tau_I(x')} \; [a \in I]$$

$$\frac{x \xrightarrow{\tau} x'}{\tau_I(x) \xrightarrow{\tau} \tau_I(x')} \qquad \frac{x \downarrow_t}{\tau_I(x) \downarrow_t} \qquad \frac{x \rightsquigarrow_t}{\tau_I(x) \rightsquigarrow_t}$$

Some typical examples revisited (I)

$$\tau_{\{b\}}(a^{@1}.b^{@2}.c^{@4}.0^{@5})$$

$$\tau_{\{b\}}(a^{@1}.b^{@3}.c^{@4}.0^{@5})$$

$$a^{@1}.c^{@4}.0^{@5}$$

Klusener: all equal

we: all equal

Some typical examples revisited (II)

$$\tau_{\{b\}}(a^{@1}.(b^{@2}.(c^{@3}.0^{@4} + d^{@3}.0^{@4}) + d^{@3}.0^{@4}))$$

$$a^{@1}.(c^{@3}.0^{@4} + d^{@3}.0^{@4})$$

Klusener: different

we: equal

Some typical examples revisited (III)

$$\tau_{\{b\}}(a^{@1}.(b^{@2}.c^{@3}.0^{@4} + d^{@3}.0^{@4}))$$

$$\tau_{\{b\}}(a^{@1}.(c^{@3}.0^{@4} + b^{@2}.d^{@3}.0^{@4}))$$

Klusener: equal

we: different

Some typical examples revisited (IV)

$$\tau_{\{b\}}(a^{@1}.(b^{@3}.0^{@4} + c^{@2}.0^{@4}))$$

$$a^{@1}.(0^{@4} + c^{@2}.0^{@4})$$

Klusener: equal

we: different

Equivalence

A symmetric binary relation R on closed terms is a *timed branching bisimulation* relation if for all r and s with $(r, s) \in R$, we have:

1. if $r \xrightarrow{a}_t r'$, then there exist s^* and s' such that $s \xrightarrow{\tau}{}^* s^* \xrightarrow{a}_t s'$, $(r, s^*) \in R$ and $(r', s') \in R$;
2. if $r \xrightarrow{\tau} r'$, then there exist s^* and s' such that $s \xrightarrow{\tau}{}^* s^* \xrightarrow{(\tau)} s'$, $(r, s^*) \in R$ and $(r', s') \in R$;
3. if $r \downarrow_t$, then there exists s^* such that $s \xrightarrow{\tau}{}^* s^* \downarrow_t$ and $(r, s^*) \in R$;
4. if $r \leadsto_t$, then there exists s^* such that $s \xrightarrow{\tau}{}^* s^* \leadsto_t$.

$$p \xrightarrow{(\tau)} q \quad \triangleq \quad p \xrightarrow{\tau} q \text{ or } p = q$$

Equivalence - Root condition

If R is a timed branching bisimulation relation, we say that the pair (p, q) satisfies the *root condition* w.r.t. R if

1. if $p \xrightarrow{a}_t p'$, then there exists a q' such that $q \xrightarrow{a}_t q'$ and $(p', q') \in R$;
2. if $p \xrightarrow{\tau} p'$, then there exists a q' such that $q \xrightarrow{\tau} q'$ and $(p', q') \in R$;
3. if $p \downarrow_t$, then $q \downarrow_t$;
4. if $p \rightsquigarrow_t$, then $q \rightsquigarrow_t$.

$p \underline{\leftrightarrow}_{\mathrm{rb}}\, q \triangleq$ there is a timed branching bisimulation relation R relating p and q such that (p, q) and (q, p) satisfy the root condition w.r.t. R.

Properties of $\underline{\leftrightarrow}_{\mathrm{rb}}$

Theorem
Timed rooted branching bisimilarity is an equivalence relation.

Theorem
Timed rooted branching bisimilarity is a congruence for all operators from the signature of the process algebra.

Theorem
Our notion is incomparable with Klusener's.

Axiomatisation

(A1)	$x + y = y + x$
(A2)	$(x + y) + z = x + (y + z)$
(A3)	$x + x = x$
(WT)	$a^{@t}.x = a^{@t}.t \gg x$
(A6a)	$0^{@t} + 0^{@u} = 0^{@\max(t,u)}$
(A6b)	$u \leq t \Rightarrow 1^{@t} + 0^{@u} = 1^{@t}$
(A6c)	$u \leq t \Rightarrow a^{@t}.x + 0^{@u} = a^{@t}.x$
(A6d)	$u \leq t \Rightarrow \tau.(x + 0^{@t}) + 0^{@u} = \tau.(x + 0^{@t})$
(B)	$a^{@t}.(\tau.(x + y) + x) = a^{@t}.(x + y)$

Axiomatisation (cont'd)

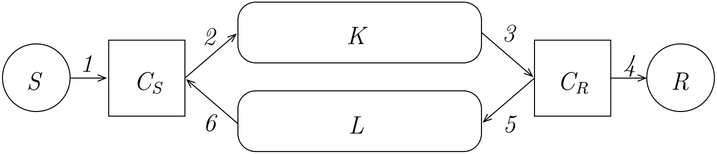
(I1)	$t \gg 0^{@u} = 0^{@\max(t,u)}$
(I2)	$u < t \Rightarrow t \gg 1^{@u} = 0^{@t}$
(I3)	$u \geq t \Rightarrow t \gg 1^{@u} = 1^{@u}$
(I4)	$u < t \Rightarrow t \gg a^{@u}.x = 0^{@t}$
(I5)	$u \geq t \Rightarrow t \gg a^{@u}.x = a^{@u}.x$
(I6)	$t \gg \tau.x = \tau.t \gg x$
(I7)	$t \gg (x + y) = t \gg x + t \gg y$

Axiomatisation (cont'd)

(H1)	$\tau_I(0^{@t}) = 0^{@t}$
(H2)	$\tau_I(1^{@t}) = 1^{@t}$
(H3)	$a \notin I \Rightarrow \tau_I(a^{@t}.x) = a^{@t}.\tau_I(t \gg x)$
(H4)	$a \in I \Rightarrow \tau_I(a^{@t}.x) = \tau.\tau_I(t \gg x)$
(H5)	$\tau_I(\tau.x) = \tau.\tau_I(x)$
(H6)	$\tau_I(x + y) = \tau_I(x) + \tau_I(y)$

PAR protocol revisited

Timed version of the Alternating Bit Protocol.



The protocol entities are a sender S, a receiver R, a forward channel K, and a backward channel L.

Channels are unreliable, but errors are observable.

PAR protocol - Timed Silent Step

$$
\begin{aligned}
X_{b,t} &= \sum_{t'} \sum_{d \in D} r_1(d)^{@t+t'} . Y_{d,b,t+t'+t_S} \\
Y_{d,b,t} &= \tau^{@t+t_K} . s_2(d)^{@t+t_K+t_R} . Z_{d,b,t+t_K+t_R+t'_R} \\
&+ \sum_{k \le t_K} \tau^{@t+k} . Y_{d,b,t+t'_S} \\
Z_{d,b,t} &= \tau^{@t+t_L} . X_{\overline{b},t+t_L} + \sum_{l \le t_L} \tau^{@t+l} . U_{d,b,t+t'_S-t_K-t_R-t'_R} \\
U_{d,b,t} &= \tau^{@t+t_K} . V_{d,b,t+t_K+t'_R} + \sum_{k \le t_K} \tau^{@t+k} . U_{d,b,t+t'_S} \\
V_{d,b,t} &= \tau^{@t+t_L} . X_{\overline{b},t+t_L} + \sum_{l \le t_L} \tau^{@t+l} . U_{d,b,t+t'_S-t_K-t'_R}
\end{aligned}
$$

PAR protocol - Untimed Silent Step

$$
\begin{aligned}
X_{b,t} &= \sum_{t'} \sum_{d \in D} r_1(d)^{@t+t'} . Y_{d,b,t+t'+t_S} \\
Y_{d,b,t} &= \tau . s_2(d)^{@t+t_k+t_R} . U'_{d,b,t,t_R} + \tau . Y_{d,b,t+t'_S} \\
U'_{d,b,t,u} &= \tau . X_{\overline{b},t+t_K+u+t'_R+t_L} + \tau . U'_{d,b,t+t'_S,0}
\end{aligned}
$$

Outline

Introduction to Timed Process Algebra
Abstraction in Timed Process Algebra
Some typical examples
Illustration: PAR protocol

Untimed Silent Step
Some typical examples revisited
Axiomatisation
Illustration: PAR protocol revisited

Concluding Remarks

Drawbacks Untimed Silent Step

Recall that Klusener has

$$\tau_{\{b\}}(a^{@1}.(b^{@3}.0^{@4} + c^{@2}.0^{@4}))$$

$$\underline{\leftrightarrow}_{\mathrm{rb}}$$

$$a^{@1}.(0^{@4} + c^{@2}.0^{@4})$$

The motivation is that the occurrence or absence of c at time 2 determines the choice and not so much the silent step (at time 3).

Drawbacks Untimed Silent Step (cont'd)

We do not have the distribution of the abstraction operator over sequential and parallel composition.
Consider $p = a^{@2}.1^{@2}$ and $q = b^{@1}.1^{@3}$. Let $I = \{b\}$. Now,

$$\tau_I(p \cdot q) \;\underline{\leftrightarrow}_{\mathrm{rb}}\; \tau_I(a^{@2}.0^{@2}) \;\underline{\leftrightarrow}_{\mathrm{rb}}\; a^{@2}.0^{@2}$$

whereas

$$\tau_I(p) \cdot \tau_I(q) \;\underline{\leftrightarrow}_{\mathrm{rb}}\; (a^{@2}.1^{@2}) \cdot (\tau.1^{@3}) \;\underline{\leftrightarrow}_{\mathrm{rb}}\; a^{@2}.1^{@3}$$

Current/Future Work

Current:

- Notion of equivalence that avoids the drawback of the untimed silent step

Future:

- Investigate other timed settings (especially two-phase models)
- Extensions with other operators (sequential, parallel, etc.)
- Practical suitability: more case studies
- Relationship with untimed process algebras

Thank you for your attention.