Causal Reversibility for Timed Process Calculi with Lazy/Eager Durationless Actions and Time Additivity

Claudio Antares Mezzina

In collaboration with Marco Bernardo

~~FORMATS 2023 / CONFEST @Antwerp~~
NiRvAna meeting @ Udine

Timed Systems - design choices

We address reversibility of real-time systems

Several design choices - **Durationless actions Vs durational actions**

- Durationless actions
    - actions are instantaneous events and time passes in between them
    - functional behaviour and time are orthogonal
- durational actions
    - every action takes a certain amount of time to be performed and time passes only due to action execution
    - functional behavior and time are integrated

Timed Systems - design choices

Several design choices - **Relative time Vs Absolute time**

- Relative time
  - Each timestamp refers to the time instant of the previous observation
- Absolute time
  - all timestamps refer to the starting time of the system execution

Timed Systems - design choices

Several design choices - **Global clock Vs Local clocks**

- Global clock
    - a single clock governs time passing
- Local clocks
    - several clocks associated with the various system parts elapse independent of each other

Timed Systems - design choices

Several interpretations of action execution in terms of whether and when the execution can be delayed

- Eagerness
    - actions must be performed as soon as they become enabled
    - e.g. actions cannot be delayed
- Laziness
    - actions can be delayed arbitrarily long before they are executed
- Maximal progress
    - enabled actions can be delayed unless they are independent of the external environment (e.g., taus)

Our design choices

We study reversibility in a timed calculus with

- Durationless actions VS durational actions
- Relative time VS absolute time
- Global clock VS local clocks

with

eager and lazy interpretation and time additivity

We start from a simple calculus inspired by Moller & Tofts [MT90]

Causal Consistent reversibility

- How you can undo a computation?
- In a sequential setting this is straightforward: you start undoing for the last action
- In a concurrent/distributed setting there is no clear definition of last action
    - We can consider as last action any action which has no consequences (e.g., it has not caused anything)
    - Hence an action can be undone provided that its consequences are undone beforehand
    - Essentially any reached state is a state that can be reached just with forward moves
    - This idea is used in transactions/rollback schemas where the system has to get back to a consistent state

Reversibility in Concurrent System

Calculi

Reversible Communicating System (RCCS) Danos&Krivine

- Use of explicit memories to keep track of past events
- Suitable for complex languages (e.g., scales with pi-calculus, Erlang)
- Give the first notion of causally consistent reversibility
- **Won CONCUR23 test of time award**

CCS with communication keys (CCSK) Phillips&Ulidowski

- History information directly recorderded into the term
- Use of keys to keep track of synchronisations
- Suitable for CCS-like languages with LTSs

Example

$$a.P + b.Q \xrightarrow{a} P$$

After the computation, we loose information about

- The performed action a
- The other branch b.Q

CCSK

$$\underline{a.P + b.Q} \xrightarrow{a[i]} \boxed{a[i]}\,P + \boxed{b.Q} \xrightarrow{\underline{a[i]}} a.P + b.Q$$

No need of extra memories

History information directly in the term

The two reversible CCSs have been shown to be equivalent LMM2021

RTPC: reversible timed process calculus

- A simple extension of CCS with time prefix **(n).P**
- Synchronisation à la CSP
- (n).P acts as P after n time units
- Reversing à la CCSK

$$\boxed{\begin{array}{lcl} P, Q & ::= & \underline{0} \mid a\,.\,P \mid (n)\,.\,P \mid P + Q \mid P \parallel_L Q \\ R, S & ::= & P \mid a[i]\,.\,R \mid (n)^{[i]}\,.\,R \mid \langle n^{i} \rangle\,.\,R \mid R + S \mid R \parallel_L S \end{array}}$$

Past action prefix **Past time prefix** **Dynamic delay prefix**

$\delta(n,i)$ denote $(n)^{[i]}$ or $\langle n^{i} \rangle$

std(R) = R has no history

npa(R) = R has not executed actions

RTPC - action semantics

$$(\textsc{Act1})\ \frac{\mathtt{std}(R)}{a\,.\,R \xrightarrow{a[i]}_{\mathsf{a}} a[i]\,.\,R}$$

$$(\textsc{Act1}^{\bullet})\ \frac{\mathtt{std}(R)}{a[i]\,.\,R \overset{a[i]}{\dashrightarrow}_{\mathsf{a}} a\,.\,R}$$

$$(\textsc{Act2})\ \frac{R \xrightarrow{b[j]}_{\mathsf{a}} R' \quad j \neq i}{a[i]\,.\,R \xrightarrow{b[j]}_{\mathsf{a}} a[i]\,.\,R'}$$

$$(\textsc{Act2}^{\bullet})\ \frac{R \overset{b[j]}{\dashrightarrow}_{\mathsf{a}} R' \quad j \neq i}{a[i]\,.\,R \overset{b[j]}{\dashrightarrow}_{\mathsf{a}} a[i]\,.\,R'}$$

$$(\textsc{Act3})\ \frac{R \xrightarrow{b[j]}_{\mathsf{a}} R'}{\underline{\delta(n,i)}\,.\,R \xrightarrow{b[j]}_{\mathsf{a}} \delta(n,i)\,.\,R'}$$

$$(\textsc{Act3}^{\bullet})\ \frac{R \overset{b[j]}{\dashrightarrow}_{\mathsf{a}} R'}{\delta(n,i)\,.\,R \overset{b[j]}{\dashrightarrow}_{\mathsf{a}} \delta(n,i)\,.\,R'}$$

$$(\textsc{Cho})\ \frac{R \xrightarrow{a[i]}_{\mathsf{a}} R' \quad \mathtt{npa}(S)}{R + S \xrightarrow{a[i]}_{\mathsf{a}} R' + S}$$

$$(\textsc{Cho}^{\bullet})\ \frac{R \overset{a[i]}{\dashrightarrow}_{\mathsf{a}} R' \quad \mathtt{npa}(S)}{R + S \overset{a[i]}{\dashrightarrow}_{\mathsf{a}} R' + S}$$

$$(\textsc{Par})\ \frac{R \xrightarrow{a[i]}_{\mathsf{a}} R' \quad a \notin L \quad i \notin \mathtt{keys}_{\mathsf{a}}(S)}{R \parallel_L S \xrightarrow{a[i]}_{\mathsf{a}} R' \parallel_L S}$$

$$(\textsc{Par}^{\bullet})\ \frac{P \overset{a[i]}{\dashrightarrow}_{\mathsf{a}} P' \quad a \notin L \quad i \notin \mathtt{keys}_{\mathsf{a}}(S)}{R \parallel_L S \overset{a[i]}{\dashrightarrow}_{\mathsf{a}} R' \parallel_L S}$$

$$(\textsc{Coo})\ \frac{R \xrightarrow{a[i]}_{\mathsf{a}} R' \quad S \xrightarrow{a[i]}_{\mathsf{a}} S' \quad a \in L}{R \parallel_L S \xrightarrow{a[i]}_{\mathsf{a}} R' \parallel_L S'}$$

$$(\textsc{Coo}^{\bullet})\ \frac{R \overset{a[i]}{\dashrightarrow}_{\mathsf{a}} R' \quad S \overset{a[i]}{\dashrightarrow}_{\mathsf{a}} S' \quad a \in L}{R \parallel_L S \overset{a[i]}{\dashrightarrow}_{\mathsf{a}} R' \parallel_L S'}$$

RTPC - time semantics 1/2

δ(n,i) denote $(n)^{[i]}$ or $\langle n^i \rangle$

std(R) = R has no history

npa(R) = R has not executed actions

Laziness

Maximal Progress

(IDLING1) $\underline{0} \xrightarrow{(n)^{[i]}}_{\mathtt{d}} \langle n^i \rangle . \underline{0}$

(IDLING2) $$\frac{\mathtt{std}(R)}{a . R \xrightarrow{(n)^{[i]}}_{\mathtt{d}} \langle n^i \rangle . a . R}$$

(IDLING3) $$\frac{\mathtt{std}(R) \quad a \neq \tau}{a . R \xrightarrow{(n)^{[i]}}_{\mathtt{d}} \langle n^i \rangle . a . R}$$

(DELAY1) $$\frac{\mathtt{std}(R)}{(n) . R \xrightarrow{(n)^{[i]}}_{\mathtt{d}} (n)^{[i]} . R}$$

(DELAY2) $$\frac{R \xrightarrow{(n)^{[j]}}_{\mathtt{d}} R'}{a[i] . R \xrightarrow{(n)^{[j]}}_{\mathtt{d}} a[i] . R'}$$

(DELAY3) $$\frac{R \xrightarrow{(m)^{[j]}}_{\mathtt{d}} R' \quad j \neq i}{\delta(n,i) . R \xrightarrow{(m)^{[j]}}_{\mathtt{d}} \delta(n,i) . R'}$$

(IDLING1$^\bullet$) $\langle n^i \rangle . \underline{0} \overset{(n)^{[i]}}{\dashrightarrow}_{\mathtt{d}} \underline{0}$

(IDLING2$^\bullet$) $$\frac{\mathtt{std}(R)}{\langle n^i \rangle . a . R \overset{(n)^{[i]}}{\dashrightarrow}_{\mathtt{d}} a . R}$$

(IDLING3$^\bullet$) $$\frac{\mathtt{std}(R) \quad a \neq \tau}{\langle n^i \rangle . a . R \overset{(n)^{[i]}}{\dashrightarrow}_{\mathtt{d}} a . R}$$

(DELAY1$^\bullet$) $$\frac{\mathtt{std}(R)}{(n)^{[i]} . R \overset{(n)^{[i]}}{\dashrightarrow}_{\mathtt{d}} (n) . R}$$

(DELAY2$^\bullet$) $$\frac{R \overset{(n)^{[j]}}{\dashrightarrow}_{\mathtt{d}} R'}{a[i] . R \overset{(n)^{[j]}}{\dashrightarrow}_{\mathtt{d}} a[i] . R'}$$

(DELAY3$^\bullet$) $$\frac{R \overset{(m)^{[j]}}{\dashrightarrow}_{\mathtt{d}} R' \quad j \neq i}{\delta(n,i) . R \overset{(m)^{[j]}}{\dashrightarrow}_{\mathtt{d}} \delta(n,i) . R}$$

RTPC - time semantics 2/2

Time Additivity

$$(\textsc{TAdd1})\ \frac{R \xrightarrow{(m)^{[j]}}_{\mathtt{d}} R' \quad \mathtt{std}(R) \quad j \neq i}{(n)\,.\,R \xrightarrow{(n+m)^{[i]}}_{\mathtt{d}} (n)^{[i]}.\,R'} \qquad (\textsc{TAdd1}^{\bullet})\ \frac{R \overset{(m)^{[j]}}{\dashrightarrow}_{\mathtt{d}} R' \quad \mathtt{std}(R') \quad j \neq i}{(n)^{[i]}.\,R \overset{(n+m)^{[i]}}{\dashrightarrow}_{\mathtt{d}} (n)\,.\,R'}$$

$$(\textsc{TAdd2})\ \frac{\mathtt{std}(R) \quad n = n_1 + n_2}{(n)\,.\,R \xrightarrow{(n_1)^{[i]}}_{\mathtt{d}} (n_1)^{[i]}.\,(n_2)\,.\,R} \qquad (\textsc{TAdd2}^{\bullet})\ \frac{\mathtt{std}(R) \quad n = n_1 + n_2}{(n_1)^{[i]}.\,(n_2)\,.\,R \overset{(n_1)^{[i]}}{\dashrightarrow}_{\mathtt{d}} (n)\,.\,R}$$

$$(\textsc{TCho1})\ \frac{\begin{array}{c} R \xrightarrow{(n)^{[i]}}_{\mathtt{d}} R' \quad S \xrightarrow{(n)^{[i]}}_{\mathtt{d}} S' \\ \mathtt{npa}(R+S) \end{array}}{R+S \xrightarrow{(n)^{[i]}}_{\mathtt{d}} R'+S'} \qquad (\textsc{TCho1}^{\bullet})\ \frac{\begin{array}{c} R \overset{(n)^{[i]}}{\dashrightarrow}_{\mathtt{d}} R' \quad S \overset{(n)^{[i]}}{\dashrightarrow}_{\mathtt{d}} S' \\ \mathtt{npa}(R+S) \end{array}}{R+S \overset{(n)^{[i]}}{\dashrightarrow}_{\mathtt{d}} R'+S'}$$

$$(\textsc{TCho2})\ \frac{\begin{array}{c} R \xrightarrow{(n)^{[i]}}_{\mathtt{d}} R' \\ \neg\mathtt{npa}(R) \quad \mathtt{npa}(S) \end{array}}{R+S \xrightarrow{(n)^{[i]}}_{\mathtt{d}} R'+S}$$

Time Determinism

$$(\textsc{TCho2}^{\bullet})\ \frac{\begin{array}{c} R \overset{(n)^{[i]}}{\dashrightarrow}_{\mathtt{d}} R' \\ \neg\mathtt{npa}(R) \quad \mathtt{npa}(S) \end{array}}{R+S \overset{(n)^{[i]}}{\dashrightarrow}_{\mathtt{d}} R'+S}$$

$$(\textsc{TCoo})\ \frac{R \xrightarrow{(n)^{[i]}}_{\mathtt{d}} R' \quad S \xrightarrow{(n)^{[i]}}_{\mathtt{d}} S'}{R \parallel_L S \xrightarrow{(n)^{[i]}}_{\mathtt{d}} R' \parallel_L S'} \qquad (\textsc{TCoo}^{\bullet})\ \frac{R \overset{(n)^{[i]}}{\dashrightarrow}_{\mathtt{d}} R' \quad S \overset{(n)^{[i]}}{\dashrightarrow}_{\mathtt{d}} S'}{R \parallel_L S \overset{(n)^{[i]}}{\dashrightarrow}_{\mathtt{d}} R' \parallel_L S'}$$

Dynamic delay operator ?

$$\begin{tikzcd}
 & & a\,.\,(n) \parallel_{\emptyset} (n) \arrow[dl] \arrow[dr] & & \\
 & a[i]\,.\,(n) \parallel_{\emptyset} (n) \arrow[ur, bend left] \arrow[ddl] & & a\,.\,(n) \parallel_{\emptyset} (n)^{[j]} \arrow[dd] & \\
 & & & & a[i]\,.\,(n)^{[k]} \parallel_{\emptyset} (n) \\
a[i]\,.\,(n)^{[j]} \parallel_{\emptyset} (n)^{[j]} \arrow[uur, bend left] & & & a[i]\,.\,(n) \parallel_{\emptyset} (n)^{[j]} \arrow[d] & \\
 & & & a[i]\,.\,(n)^{[k]} \parallel_{\emptyset} (n)^{[j]} \arrow[uur, bend right] &
\end{tikzcd}$$

???? This state cannot be reached by just forward moves

Dynamic delay operator LBMY22

$$\begin{tikzcd}
 & a \,.\, (n) \parallel_{\emptyset} (n) \arrow[dl] \arrow[dr] & & \\
a[i] \,.\, (n) \parallel_{\emptyset} (n) \arrow[ur, bend left] \arrow[d] & & \langle n^{j} \rangle \,.\, a \,.\, (n) \parallel_{\emptyset} (n)^{[j]} \arrow[d] & \\
a[i] \,.\, (n)^{[j]} \parallel_{\emptyset} (n)^{[j]} \arrow[u, bend left] & & \langle n^{j} \rangle \,.\, a[i] \,.\, (n) \parallel_{\emptyset} (n)^{[j]} \arrow[d] & {} \\
 & & \langle n^{j} \rangle \,.\, a[i] \,.\, (n)^{[k]} \parallel_{\emptyset} (n)^{[j]} \arrow[ur, bend right, "\times" marking] &
\end{tikzcd}$$

j cannot be undone as it
$i < k$

Properties

Proposition 1 (time determinism). *Let* $R, S_1, S_2 \in \mathbb{P}$, $n \in \mathbb{N}_{>0}$, $i_1, i_2 \in \mathcal{K}$, *and* $j \in \mathcal{K}$ *be not occurring associated with past delays in* S_1 *and* S_2. *Then:*

– *If* $R \xrightarrow{(n)^{[i_1]}}_{\mathtt{d}} S_1$ *and* $R \xrightarrow{(n)^{[i_2]}}_{\mathtt{d}} S_2$, *then* $S_1\{^j/_{i_1}\} = S_2\{^j/_{i_2}\}$.
– *If* $R \overset{(n)^{[i_1]}}{\dashrightarrow}_{\mathtt{d}} S_1$ *and* $R \overset{(n)^{[i_2]}}{\dashrightarrow}_{\mathtt{d}} S_2$, *then* $S_1 = S_2$. $\blacksquare$

Proposition 2 (time additivity). *Let* $R, R', S, S' \in \mathbb{P}$, $n, h \in \mathbb{N}_{>0}$, $i \in \mathcal{K}$, *and* $m_1, \ldots, m_h \in \mathbb{N}_{>0}$ *be such that* $\sum_{1 \le l \le h} m_l = n$. *Then:*

– $R \xrightarrow{(n)^{[i]}}_{\mathtt{d}} S$ *iff* $R \xrightarrow{(m_1)^{[i_1]}}_{\mathtt{d}} \ldots \xrightarrow{(m_h)^{[i_h]}}_{\mathtt{d}} S'$.
– $R \overset{(n)^{[i]}}{\dashrightarrow}_{\mathtt{d}} S$ *iff* $R' \overset{(m_1)^{[i_1]}}{\dashrightarrow}_{\mathtt{d}} \ldots \overset{(m_h)^{[i_h]}}{\dashrightarrow}_{\mathtt{d}} S$. $\blacksquare$

Axioms of reversibility

In LPU20 a general framework to prove causal consistency is given

- On a LTS with an independence relation and on which loop lemma holds
- **Square property** (SP), **well-foundness** (WF) and **backward transition independence** (BTI)

  In RTPC

  - Loop Lemma: Any transition can be undone
    - does not hold
  - Square property: Concurrent transitions can be swapped
    - does hold
  - WF: History information is finite
    - does hold
  - BTI holds as long as at least one of the two coinitial backward transitions is not a delay transition

Loop Property (weak)

Proposition 3 (loop property). *Let* $R, S \in \mathbb{P}$, $a[i], (n)^{[i]} \in \mathcal{L}$, $h \in \mathbb{N}_{>0}$, *and* $m_1, \ldots, m_h \in \mathbb{N}_{>0}$ *be such that* $\sum_{1 \leq l \leq h} m_l = n$. *Then:*

- $R \xrightarrow{a[i]}_{\mathtt{a}} S$ *iff* $S \overset{a[i]}{\dashrightarrow}_{\mathtt{a}} R$.
- *If* $R \xrightarrow{(n)^{[i]}}_{\mathtt{d}} S$ *then* $S \overset{(n)^{[i]}}{\dashrightarrow}_{\mathtt{d}} R$.
- $R \xrightarrow{(n)^{[i]}}_{\mathtt{d}} S$ *iff* $S \overset{(m_1)^{[i_1]}}{\dashrightarrow}_{\mathtt{d}} \ldots \overset{(m_h)^{[i_h]}}{\dashrightarrow}_{\mathtt{d}} R$.
- $R \overset{(n)^{[i]}}{\dashrightarrow}_{\mathtt{d}} S$ *iff* $S \xrightarrow{(m_1)^{[i_1]}}_{\mathtt{d}} \ldots \xrightarrow{(m_h)^{[i_h]}}_{\mathtt{d}} R$. ■

$$\begin{tikzcd}
 & (2)\,.\,a\,.\,\underline{0} + (2)\,.\,b\,.\,\underline{0} \arrow[dl, bend right, "(1)^{[i]}"'] \arrow[dr, bend left, "(2)^{[k]}"] & \\
(1)^{[i]}.\,(1)\,.\,a\,.\,\underline{0} + (1)^{[i]}.\,(1)\,.\,b\,.\,\underline{0} \arrow[ur, dashed, "(1)^{[i]}"'] \arrow[d, bend right, "(1)^{[j]}"'] & & (2)^{[k]}.\,a\,.\,\underline{0} + (2)^{[k]}.\,b\,.\,\underline{0} \arrow[ul, dashed, "(2)^{[k]}"] \\
(1)^{[i]}.\,(1)^{[j]}.\,a\,.\,\underline{0} + (1)^{[i]}.\,(1)^{[j]}.\,b\,.\,\underline{0} \arrow[u, dashed, bend right, "(1)^{[j]}"'] \arrow[uur, dashed, "(2)^{[i]}"'] & &
\end{tikzcd}$$

No BTI

No classical loop lemma

Backward triangularity

Lemma 3 (backward triangularity). *Let* $R \in \mathbb{P}$. *Whenever* $R \overset{(n)^{[i]}}{\dashrightarrow}_{\mathtt{d}} S_1$ *and* $R \overset{(m)^{[j]}}{\dashrightarrow}_{\mathtt{d}} S_2$ *with* $m > n$, *then* $S_1 \overset{(m-n)^{[k]}}{\dashrightarrow}_{\mathtt{d}} S_2$. ■

$$\begin{tikzcd}
 & (2)\,.\,a\,.\,\underline{0} + (2)\,.\,b\,.\,\underline{0} \arrow[dl, bend right, "(1)^{[i]}"'] \arrow[dr, bend left, "(2)^{[k]}"] & \\
(1)^{[i]}.\,(1)\,.\,a\,.\,\underline{0} + (1)^{[i]}.\,(1)\,.\,b\,.\,\underline{0} \arrow[ur, dashed, "(1)^{[i]}"'] \arrow[d, bend right, "(1)^{[j]}"'] & & (2)^{[k]}.\,a\,.\,\underline{0} + (2)^{[k]}.\,b\,.\,\underline{0} \arrow[ul, dashed, "(2)^{[k]}"] \\
(1)^{[i]}.\,(1)^{[j]}.\,a\,.\,\underline{0} + (1)^{[i]}.\,(1)^{[j]}.\,b\,.\,\underline{0} \arrow[u, dashed, bend right, "(1)^{[j]}"'] \arrow[uur, dashed, bend right, "(2)^{[i]}"'] & &
\end{tikzcd}$$

Causal equivalence

Definition 2 (causal equivalence). Causal equivalence $\asymp$ *is the smallest equivalence relation over paths that is closed under composition and satisfies the following:*

1. $\theta_1\theta_2' \asymp \theta_2\theta_1'$ *for any two coinitial concurrent action transitions* $\theta_1 : R \xmapsto{a[i]} R_1$ *and* $\theta_2 : R \xmapsto{b[j]} R_2$ *and any two cofinal action transitions* $\theta_2' : R_1 \xmapsto{b[j]} S$ *and* $\theta_1' : R_2 \xmapsto{a[i]} S$ *respectively composable with the previous ones.*
2. $\theta\overline{\theta} \asymp \epsilon$ *and* $\overline{\theta}\theta \asymp \epsilon$ *for any transition* θ*.*
3. $\omega_1\overline{\omega_2} \asymp \epsilon$ *and* $\overline{\omega_2}\omega_1 \asymp \epsilon$ *for any two coinitial and cofinal forward paths* ω_1 *and* ω_2 *with delay transitions only such that* $\texttt{time}(\omega_1) = \texttt{time}(\omega_2)$. ■

Properties

Lemma 5 (parabolic lemma). *For any path ω, there exist two forward paths ω_1 and ω_2 such that $\omega \asymp \overline{\omega_1}\omega_2$ and $|\omega_1| + |\omega_2| \leq |\omega|$.* ■

We could not exploit the axioms of Lanese, Phillips & Ulidowski to prove PB

Theorem 1 (causal consistency). *Let ω_1 and ω_2 be two paths. Then $\omega_1 \asymp \omega_2$ iff ω_1 and ω_2 are both coinitial and cofinal.* ■

We could not exploit the axioms of Lanese, Phillips & Ulidowski to prove CC

PB + WF (+ ~~looplemma~~) = CC

Timeout operator [MT90]

Under maximal progress, we can encode a timeout operator Timeout(P, Q, n)

- It allows the process P to communicate with the environment within n units of time
- After this time has passed, and P has not communicated yet, the process Q takes control

$$\textsc{Timeout}(P, Q, n) = P + (n) \,.\, \tau \,.\, Q$$

Erlang receive

```
process_A() ->
  receive
    X -> handleMsg()
     after 50 ->
       handleTimeout()
  end end.
```

```
process_B(Pid) ->
  timer:sleep(100),
  Pid! Msg  end.

PidA=spawn(?MODULE, process_A , []) ,
spawn(?MODULE, process_B , [PidA]).
```

Process A awaits a message from the environment (B)

- If the message is received within 50 ms then handleMsg() is called
- Otherwise handleTimeout() is called

RTPC example

$$\begin{aligned} A &= \textsc{Timeout}(a\,.\,P, Q, 50) \\ B &= (100)\,.\,a\,.\,\underline{0} \end{aligned}$$

P is the process encoding handleMsg()

Q is the process encoding handleTimeout()

$$\begin{aligned} A \parallel_{\{a\}} B &\xrightarrow{(50)^{[i]}}_{\mathtt{d}} (a\,.\,P + (50)^{[i]}.\,\tau\,.\,Q) \parallel_{\{a\}} ((50)^{[i]}.\,(50)\,.\,a\,.\,\underline{0}) \\ &\xrightarrow{\tau[j]}_{\mathtt{a}} (a\,.\,P + (50)^{[i]}.\,\tau[j]\,.\,Q) \parallel_{\{a\}} ((50)^{[i]}.\,(50)\,.\,a\,.\,\underline{0}) \end{aligned}$$

Conclusions

- We have studied reversibility in a calculus with
  - Eagerness / laziness and where time is modelled via numeric delays
  - Numeric delays are subject to time additivity
- Because of time additivity
  - Loop property is stated differently
  - BTI does not hold

As future work, we plan to investigate suitable notions of bisimilarity for RTPC

Better way to deal with dynamic delay prefix (use zone?)

Study reversibility and dense time

References

[MT90] F. Moller, C. M. N. Tofts: A Temporal Calculus of Communicating Systems. CONCUR 90

[RCCS] V. Danos, J. Krivine: Reversible Communicating Systems. CONCUR 2004

[CCSK] I. C. C. Phillips, I. Ulidowski: Reversing algebraic process calculi. J. Log. Algebraic Methods Program. 73(1-2): 70-96 (2007)

[LPU20] I. Lanese, I. Phillips, I. Ulidowski: An Axiomatic Approach to Reversible Computation. FoSSaCS 2020

[LMM2021] I. Lanese, D. Medic, C. A. Mezzina: Static versus dynamic reversibility in CCS. Acta Informatica 58(1-2): 1-34 (2021)

[BLMY24] L. Bocchi, I. Lanese, C. A. Mezzina, S. Yuen: revTPL: The Reversible Temporal Process Language. Logical Methods in Computer science 20(1):2024 (**Supersedes FORTE22 paper**)