Towards Bridging Time and Causal Reversibility

Marco Bernardo and Claudio Antares Mezzina

Dipartimento di Scienze Pure e Applicate, Università di Urbino, Italy

NiRvAna Kick off meeting @ Fano

Reversibility?

In a reversible system we can observe two directions of computation: forward and backward

Different interpretations of reversibility in different fields:

- Petri nets = the ability to return to the initial marking
- Distributed systems = the ability to return to a past consistent state
- Performances evaluation = time reversibility
- Thermodynamics = entropy is minimised when a process is nearly reversible

Our Aim

- Our focus is in the relationship between time and causal consistent reversibility
- from a process algebraic point of view

Causal Consistent Reversibility

- If something goes wrong start undoing from the last action till the “nearest” safe state
- In a distributed system it is hard to tell which was the last performed action
- A good approximation is to consider as last action any action which has no causes
- Causal Consistent reversibility relates causality with reversibility
- Concurrent/independent actions can be reverted in any order

Causal Consistent Reversibility in PA

There exist two approaches to reverse CCS

- Dynamic approach proposed by Danos & Krivine
  - uses a stack-based memory to remember all the actions
  - more suitable with calculi with reduction semantics
- Static approach proposed by Ulidowski & Phillips
  - makes all the operator static
  - very handy with CCS-like calculi with LTSs

Static Reversibility

Dynamic operators (e.g., prefixes and non-deterministic choices) are forgetful

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

After the reduction we lose informations about the discarded branch and the executed action

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

The red terms are considered decorations of the process

$$a[i].P + b.Q = C[P]$$

Time Reversibility

- Time reversibility is instrumental to develop efficient analysis methods
- Considered in the performance evaluation field
- Related to time-reversal Markov chains
- A continuous-time Markov chain is time reversible if its behaviour remains the same when the direction of time is reversed [Kelly]
    - Mostly used in the theory of queuing networks

[Kelly] Reversibility and Stochastic Networks. John Wiley & Sons, 1979.

Markov process syllabus

- A reversible process is an ergodic process
- A CTMC is time reversible

$$\text{iff } (X(t_i))_{1 \leq i \leq n} \text{ has the same joint distribution as } (X(t' - t_i))_{1 \leq i \leq n}$$

- In which case X(t) and its reversed Xr(t) are **stocastically** equivalent
- For a stationary CTMC to be time reversible it is necessary and sufficient to verify the partial balance equation for every state

$$\pi(s) \cdot q_{s,s'} = \pi(s') \cdot q_{s',s}$$

RMPC - Syntax

Reversible Markovian Process Calculus

$$P, Q \ ::= \ \mathbf{0} \mid (a, \lambda).P \mid P + Q \mid P \parallel_L Q$$

$$R, S \ ::= \ P \mid (a, \lambda)[i].R \mid R + S \mid R \parallel_L S$$

- Each action is paired with a real positive rate λ, μ
- Cooperation is done à la CSP $\parallel_L$
- No recursion (otherwise infinite state)
- From a stochastic process algebra a CTMC can be derived [Hillston]

[Hillston] J. Hillston. A Compositional Approach to Performance Modelling. Cambridge University Press, 1996.

RMPC Semantics

$\mathtt{std}(R)$ = history less process

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

$$\textsc{Act1}^\bullet\ \frac{\mathtt{std}(R)}{(a,\lambda)[i].R \overset{(a,\overline{\lambda})[i]}{\rightsquigarrow} (a,\lambda).R}$$

How do we set $\overline{\lambda}$?

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

$$\textsc{Act2}^\bullet\ \frac{R \overset{(b,\overline{\mu})[j]}{\rightsquigarrow} R' \quad j \neq i}{(a,\lambda)[i].R \overset{(b,\overline{\mu})[j]}{\rightsquigarrow} (a,\lambda)[i].R'}$$

$$\textsc{Cho}\ \frac{R \xrightarrow{(a,\lambda)[i]} R' \qquad \mathtt{std}(S)}{R + S \xrightarrow{(a,\lambda)[i]} R' + S}$$

$$\textsc{Cho}^\bullet\ \frac{R \overset{(a,\overline{\lambda})[i]}{\rightsquigarrow} R' \qquad \mathtt{std}(S)}{R + S \overset{(a,\overline{\lambda})[i]}{\rightsquigarrow} R' + S}$$

$$\textsc{Par}\ \frac{R \xrightarrow{(a,\lambda)[i]} R' \quad a \notin L \quad i \notin \mathtt{key}(S)}{R \parallel_L S \xrightarrow{(a,\lambda)[i]} R' \parallel_L S}$$

$$\textsc{Par}^\bullet\ \frac{R \overset{(a,\overline{\lambda})[i]}{\rightsquigarrow} R' \quad a \notin L \quad i \notin \mathtt{key}(S)}{R \parallel_L S \overset{(a,\overline{\lambda})[i]}{\rightsquigarrow} R' \parallel_L S}$$

$$\textsc{Coo}\ \frac{R \xrightarrow{(a,\lambda)[i]} R' \quad S \xrightarrow{(a,\mu)[i]} S' \quad a \in L}{R \parallel_L S \xrightarrow{(a,\lambda\cdot\mu)[i]} R' \parallel_L S'}$$

$$\textsc{Coo}^\bullet\ \frac{R \overset{(a,\overline{\lambda})[i]}{\rightsquigarrow} R' \quad S \overset{(a,\overline{\mu})[i]}{\rightsquigarrow} S' \quad a \in L}{R \parallel_L S \overset{(a,\overline{\lambda}\cdot\overline{\mu})[i]}{\rightsquigarrow} R' \parallel_L S'}$$

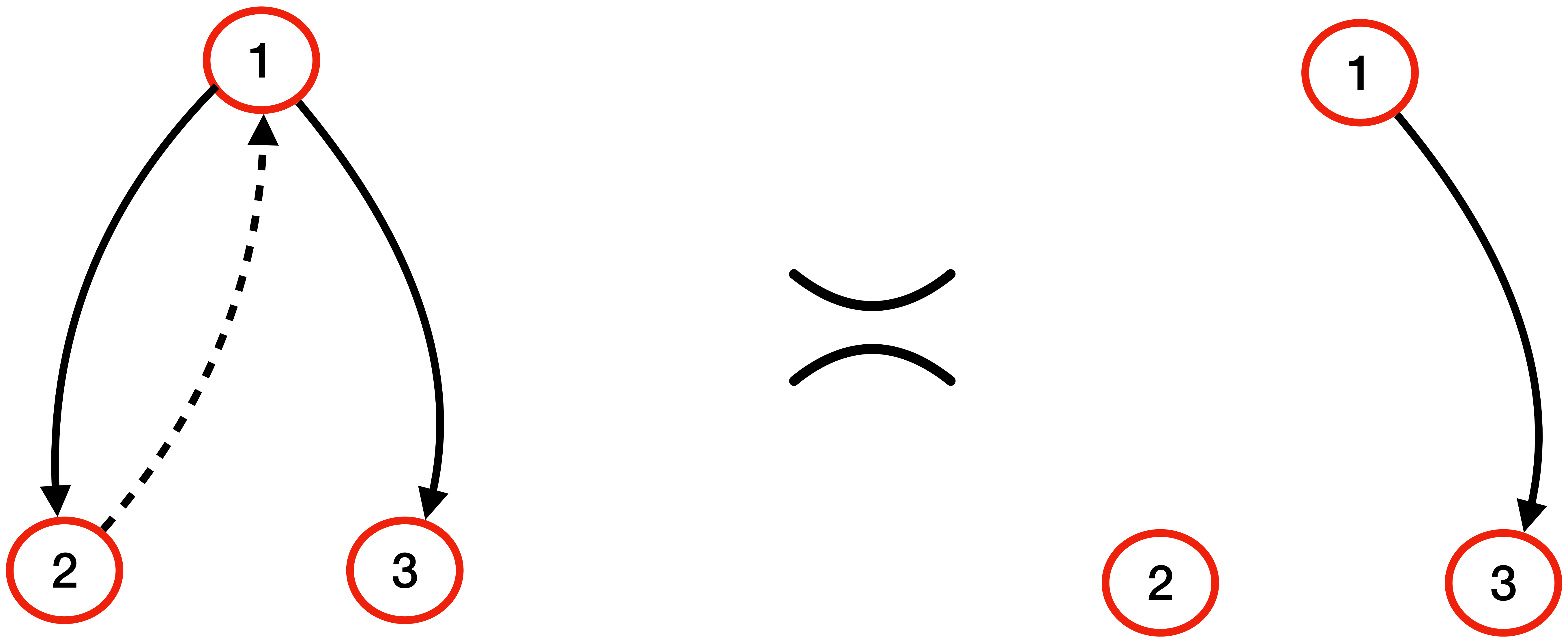
RMPC results 1/2

To prove causal consistency there is no need to specify how $\overline{\lambda}$ is calculated

Lemma 1 (loop lemma). *Let $R \in \mathcal{P}$ be a reachable process. Then $R \xrightarrow{(a,\lambda)[i]} S$ iff $S \overset{(a,\overline{\lambda})[i]}{\rightsquigarrow} R$.*

Theorem 1 (causal consistency). *Let ω_1 and ω_2 be coinitial computations. Then $\omega_1 \asymp \omega_2$ iff ω_1 and ω_2 are cofinal too.*

Explanation of causal consistency on traces



RMPC results 2/2

Markov Chain associated to process R

Theorem 2 (time reversibility). *Let $R \in \mathcal{P}$ be an initial process. If every backward rate is equal to the corresponding forward rate, then $\mathcal{M}[\![R]\!]$ is time reversible.*

Corollary 3 (product form). *Let $R, S \in \mathcal{P}$ be initial processes and $L \subseteq \mathcal{A}$. If every backward rate is equal to the corresponding forward rate and the set of states $\mathcal{S}$ of $\mathcal{M}[\![R \parallel_L S]\!]$ is equal to $\mathcal{S}_R \times \mathcal{S}_S$ where $\mathcal{S}_R$ is the set of states of $\mathcal{M}[\![R]\!]$ and $\mathcal{S}_S$ is the set of states of $\mathcal{M}[\![S]\!]$, then $\pi(r, s) = \pi_R(r) \cdot \pi_S(s)$ for all $(r, s) \in \mathcal{S}_R \times \mathcal{S}_S$.*

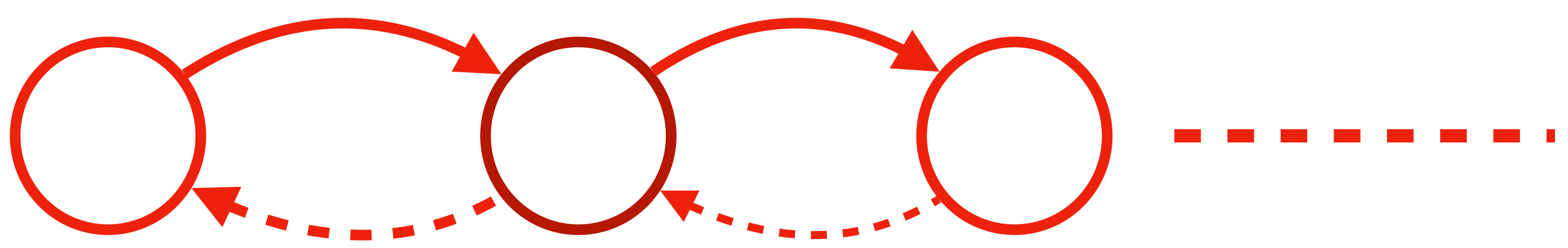
RMPC results sum up

- Reversibility induced by RMPC is causal consistent (whatever the negative rate is)
- Reversibility induced by RMPC is time reversible if we set $\lambda = \overline{\lambda}$
- Hence when $\lambda = \overline{\lambda}$ reversibility in RMPC is **both** time and causal consistent

Future work

- The cooperation operator may not require communication keys
  - No communication keys = no state explosion with recursion

$$P = (a, \lambda).P$$



- Find new condition (e.g., different from rate equality) under which time and causal reversibility holds

Easter egg

Probabilistic bisimulation

- back and forth bisimulation coincides with strong bisimulation [DNMVaandrager90]
- Following this result we have shown that back and forth markovian bisimulation coincides with markovian bisimulation (e.g., the extortion of normal bisimulation with rates)
- Open question: what about Sproston&Donatelli's Backward Bisimulation

Ester egg 2

Process Algebras and Markovian Chains [Brinksma&Hermanns]

Maximal progress of TPL (and `revTPL`) has tight connection with Markov chains [3]: a behaviour $\tau.P + (\lambda).Q$ (where λ is a rate) will not be delayed since τ is instantaneously enabled. This somehow resembles the maximal progress for the timeout operator.

$$\textsc{STout}\ \frac{X \overset{\tau}{\not\to} \qquad \mathtt{std}(X) \qquad \mathtt{std}(Y)}{\lfloor X \rfloor (Y) \xrightarrow{\sigma[i]} \lfloor X \rfloor [\underrightarrow{i}](Y)}$$

$$\textsc{SynW}\ \frac{X \xrightarrow{\sigma[u]} X' \qquad Y \xrightarrow{\sigma[v]} Y' \qquad (X \parallel Y) \overset{\tau}{\not\to} \qquad \delta(u,v) = w}{X \parallel Y \xrightarrow{\sigma[w]} X' \parallel Y'}$$