PRPC: Semantics, Logics, Axioms

A Process Algebraic Theory of Reversible Computing

Marco Bernardo
University of Urbino – Italy

PRIN 2020 project *NiRvAna*

Concurrency: Nondeterminism vs. Irreversibility

- Systems composed of several interconnected computing parts that communicate by exchanging information or simply synchronizing.
- Models: shared memory, message passing, web services, cloud, ...
- Types: centralized/distributed/decentralized, static/dynamic/mobile.
- Aspects: functionality, security, reliability, performance, ...

Concurrency: Nondeterminism vs. Irreversibility

- Systems composed of several interconnected computing parts that communicate by exchanging information or simply synchronizing.
- Models: shared memory, message passing, web services, cloud, ...
- Types: centralized/distributed/decentralized, static/dynamic/mobile.
- Aspects: functionality, security, reliability, performance, ...
- Nondeterminism: the input does not uniquely define the output.
- Different advancing speeds, scheduling policies, ...

Concurrency: Nondeterminism vs. Irreversibility

- Systems composed of several interconnected computing parts that communicate by exchanging information or simply synchronizing.
- Models: shared memory, message passing, web services, cloud, . . .
- Types: centralized/distributed/decentralized, static/dynamic/mobile.
- Aspects: functionality, security, reliability, performance, . . .
- Nondeterminism: the input does not uniquely define the output.
- Different advancing speeds, scheduling policies, . . .
- What if the output does not uniquely define the input?
- Irreversibility: typical of functions that are *not invertible*.
- Example 1: conjunctions/disjunctions are irreversible.
- Example 2: negation is reversible.

Reversible Computing

- What does (ir)reversibility mean in computing?
- Well established concept in mathematics, physics, chemistry, biology: inverse relation/function/operation/formula/law/reaction ...
- Much more recent in informatics: seminal papers by Landauer in 1961 and Bennett in 1973 on IBM Journal of Research and Development.

Reversible Computing

- What does (ir)reversibility mean in computing?
- Well established concept in mathematics, physics, chemistry, biology: inverse relation/function/operation/formula/law/reaction ...
- Much more recent in informatics: seminal papers by Landauer in 1961 and Bennett in 1973 on IBM Journal of Research and Development.
- Landauer principle states that any manipulation of information that is *irreversible* – i.e., causes information loss – such as:
  - erasure/overwriting of bits
  - merging of computation paths

  must be accompanied by a corresponding *entropy increase*.
- Minimal *heat generation* due to *extra work* for standardizing signals and making them independent of their history, so that it becomes *impossible to determine the input from the output*.

- Due to Landauer principle, the logical irreversibility of a function implies the physical irreversibility of computing that function and the consequent dissipative effects.
- Experimentally verified by Bérut et al in 2012 and revisited in terms of its physical foundations by Frank in 2018.
- Every reversible computation, where no information is lost instead, may be potentially carried out without dissipating further heat.

- Due to Landauer principle, the logical irreversibility of a function implies the physical irreversibility of computing that function and the consequent dissipative effects.
- Experimentally verified by Bérut et al in 2012 and revisited in terms of its physical foundations by Frank in 2018.
- Every reversible computation, where no information is lost instead, may be potentially carried out without dissipating further heat.
- Lower energy consumption could therefore be achieved by resorting to reversible computing.
- There are many other applications of reversible computing:
  - Biochemical reaction modeling (nature).
  - Parallel discrete-event simulation (speedup).
  - Fault-tolerant computing systems (rollback).
  - Robotics and control theory (backtrack).
  - Concurrent program debugging (reproducibility).
  - Distributed algorithms (deadlock, consensus).

- Two directions of computation characterize every reversible system:
  - Forward: coincides with the normal way of computing.
  - Backward: the effects of the forward one are undone (when needed).
- How to proceed backward? Same path as the forward direction? Is the last executed action uniquely identifiable?
- Not necessarily, especially in the case of a concurrent system; e.g., causally independent paths should be deemed equivalent.

- Two directions of computation characterize every reversible system:
  - Forward: coincides with the normal way of computing.
  - Backward: the effects of the forward one are undone (when needed).
- How to proceed backward? Same path as the forward direction? Is the last executed action uniquely identifiable?
- Not necessarily, especially in the case of a concurrent system; e.g., causally independent paths should be deemed equivalent.
- Different notions of reversibility developed in different settings:
  - Causal reversibility is the capability of going back to a past state *consistently with the computational history*: an action can be undone iff all of its consequences have been undone already [DanosKrivine04].
  - Time reversibility refers to the conditions under which the stochastic behavior remains the same when the *direction of time* is reversed (quantitative models, efficient performance evaluation) [Kelly79].
  - Only recently the relationships between the two have been investigated (the former implies the latter over models based on Markov chains when certain constraints are met).

Reversibility in Process Algebra

- There are no inverse process algebraic operators!

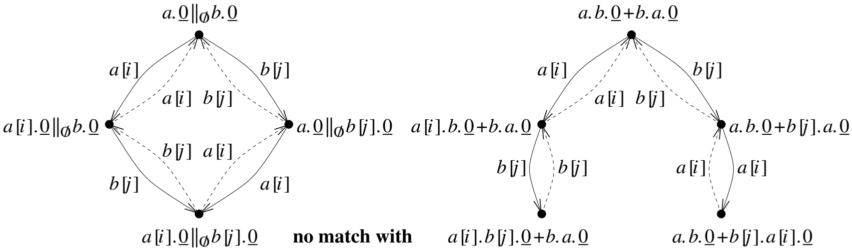
Reversibility in Process Algebra

- There are no inverse process algebraic operators!
- The dynamic approach of [DanosKrivine04] yielding RCCS uses explicit stack-based memories attached to processes to record all executed actions and all discarded subprocesses.
- A single transition relation is defined, while actions are divided into forward and backward resulting in forward and backward transitions.

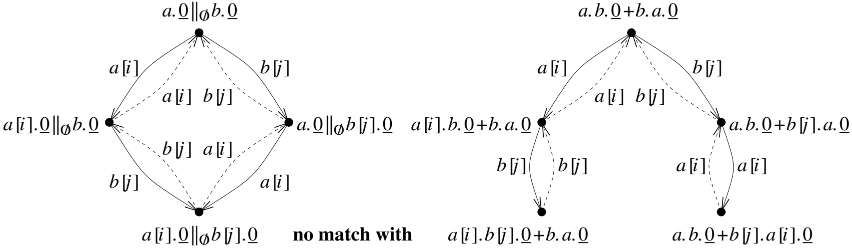
Reversibility in Process Algebra

- There are no inverse process algebraic operators!
- The dynamic approach of [DanosKrivine04] yielding RCCS uses explicit stack-based memories attached to processes to record all executed actions and all discarded subprocesses.
- A single transition relation is defined, while actions are divided into forward and backward resulting in forward and backward transitions.
- The static approach of [PhillipsUlidowski07] yielding CCSK is a method to reverse calculi by retaining within process syntax:
  - all executed actions, which are suitably decorated;
  - all dynamic operators, which are therefore treated as static.
- A forward transition relation and a backward transition relation are separately defined, labeled with communication keys so as to know who synchronized with whom when building backward transitions.

- In [PU07] forward-reverse bisimilarity has been introduced too, which is truly concurrent as it does not satisfy the expansion law of parallel composition into a choice among all possible action sequencings ($a \neq b$):



- In [PU07] forward-reverse bisimilarity has been introduced too, which is truly concurrent as it does not satisfy the expansion law of parallel composition into a choice among all possible action sequencings $(a \neq b)$:



- With back-and-forth bisimilarity [DeNicolaMontanariVaandrager90] the interleaving view can be restored as this bisimilarity is defined on computations instead of states to preserve both causality and history (one transition relation, viewed as bidirectional, outgoing/incoming).

- What are the properties of bisimilarity over reversible processes?
- Minimal process calculus tailored for reversible processes to *comparatively* study congruence, logics, and axioms for:
  - Forward-reverse bisimilarity.
  - Forward bisimilarity.
  - Reverse bisimilarity.

- What are the properties of bisimilarity over reversible processes?
- Minimal process calculus tailored for reversible processes to *comparatively* study congruence, logics, and axioms for:
  - Forward-reverse bisimilarity.
  - Forward bisimilarity.
  - Reverse bisimilarity.
- Two different kinds of bisimilarities:
  - Strong bisimilarities (all actions are treated in the same way).
  - Weak bisimilarities (abstracting from unobservable actions).

- What are the properties of bisimilarity over reversible processes?
- Minimal process calculus tailored for reversible processes to *comparatively* study congruence, logics, and axioms for:
  - Forward-reverse bisimilarity.
  - Forward bisimilarity.
  - Reverse bisimilarity.
- Two different kinds of bisimilarities:
  - Strong bisimilarities (all actions are treated in the same way).
  - Weak bisimilarities (abstracting from unobservable actions).
- Considering only sequential processes (i.e., no parallel composition) to be neutral with respect to interleaving view vs. true concurrency.
- Adding parallel composition to uniformly investigate expansion laws (relate sequential *specifications* to concurrent *implementations*).
- Characterizations via other behavioral equivalences.

- What are the properties of bisimilarity over reversible processes?
- Minimal process calculus tailored for reversible processes to *comparatively* study congruence, logics, and axioms for:
  - Forward-reverse bisimilarity.
  - Forward bisimilarity.
  - Reverse bisimilarity.
- Two different kinds of bisimilarities:
  - Strong bisimilarities (all actions are treated in the same way).
  - Weak bisimilarities (abstracting from unobservable actions).
- Considering only sequential processes (i.e., no parallel composition) to be neutral with respect to interleaving view vs. true concurrency.
- Adding parallel composition to uniformly investigate expansion laws (relate sequential *specifications* to concurrent *implementations*).
- Characterizations via other behavioral equivalences.
- Can we avoid external memories and communication keys?

PRPC – Proved Reversible Process Calculus

- Countable set $\mathcal{A}$ of actions including the unobservable action τ, renaming $\rho : \mathcal{A} \to \mathcal{A}$ s.t. $\rho(\tau) = \tau$, synchronization set $L \subseteq \mathcal{A} \setminus \{\tau\}$.
- Usually only the future behavior of processes is described.

PRPC – Proved Reversible Process Calculus

- Countable set $\mathcal{A}$ of actions including the unobservable action τ, renaming $\rho : \mathcal{A} \to \mathcal{A}$ s.t. $\rho(\tau) = \tau$, synchronization set $L \subseteq \mathcal{A} \setminus \{\tau\}$.
- Usually only the future behavior of processes is described.
- We store the past behavior in the syntax like in [PU07]:

$$P \ ::= \ \underline{0} \mid a\,.\,P \mid a^{\dagger}.\,P \mid P \llcorner \rho \urcorner \mid P + P \mid P \parallel_L P$$

- $a^{\dagger}.\,P$ executed action a, its forward continuation is inside P, and can undo a after all executed actions within P have been undone.

PRPC – Proved Reversible Process Calculus

- Countable set $\mathcal{A}$ of actions including the unobservable action τ, renaming $\rho : \mathcal{A} \to \mathcal{A}$ s.t. $\rho(\tau) = \tau$, synchronization set $L \subseteq \mathcal{A} \setminus \{\tau\}$.
- Usually only the future behavior of processes is described.
- We store the past behavior in the syntax like in [PU07]:

$$P \ ::= \ \underline{0} \mid a \,.\, P \mid a^{\dagger} .\, P \mid P \llcorner \rho \urcorner \mid P + P \mid P \parallel_{L} P$$

- $a^{\dagger} .\, P$ executed action a, its forward continuation is inside P, and can undo a after all executed actions within P have been undone.
- Single transition relation like in [DMV90] labeled just with actions.
- Therefore there is no need of communication keys [PU07], which allows for uniform action decorations like in [BoudolCastellani94].
- No need to distinguish between forward and backward actions or resort to stack-based memories [DK04].

- Initial processes: standard as all of their actions are unexecuted.

- Initial processes: standard as all of their actions are unexecuted.
- Set P of well-formed processes with unexecuted and executed actions:

$$
\begin{array}{rcl}
\mathit{wf}(\underline{0}) & & \\
\mathit{wf}(a\,.\,P') & \text{iff} & \mathit{initial}(P') \\
\mathit{wf}(a^{\dagger}.\,P') & \text{iff} & \mathit{wf}(P') \\
\mathit{wf}(P' \llcorner \rho \urcorner) & \text{iff} & \mathit{wf}(P') \\
\mathit{wf}(P_1 + P_2) & \text{iff} & (\mathit{wf}(P_1) \wedge \mathit{initial}(P_2)) \vee \\
 & & (\mathit{initial}(P_1) \wedge \mathit{wf}(P_2)) \\
\mathit{wf}(P_1 \parallel_L P_2) & \text{iff} & \mathit{wf}(P_1) \wedge \mathit{wf}(P_2)
\end{array}
$$

- Initial processes: standard as all of their actions are unexecuted.
- Set P of well-formed processes with unexecuted and executed actions:

$$\begin{array}{rcl} \mathit{wf}(\underline{0}) & & \\ \mathit{wf}(a\,.\,P') & \text{iff} & \mathit{initial}(P') \\ \mathit{wf}(a^{\dagger}\,.\,P') & \text{iff} & \mathit{wf}(P') \\ \mathit{wf}(P' \llcorner \rho \urcorner) & \text{iff} & \mathit{wf}(P') \\ \mathit{wf}(P_1 + P_2) & \text{iff} & (\mathit{wf}(P_1) \wedge \mathit{initial}(P_2)) \vee \\ & & (\mathit{initial}(P_1) \wedge \mathit{wf}(P_2)) \\ \mathit{wf}(P_1 \parallel_L P_2) & \text{iff} & \mathit{wf}(P_1) \wedge \mathit{wf}(P_2) \end{array}$$

- $\underline{0}$ is both initial and well-formed.
- Any initial process is well-formed too.
- P also contains processes that are not initial: $a^{\dagger}\,.\,b\,.\,\underline{0}$.
- Past actions can never follow future actions: $b\,.\,a^{\dagger}\,.\,\underline{0} \notin$ P.
- Alternative processes cannot be both non-initial: $a^{\dagger}\,.\,\underline{0} + b^{\dagger}\,.\,\underline{0} \notin$ P.

- Since all information needed to enable reversibility is in the syntax, action prefix and choice are made static by the semantics [PU07].
- Labeling every transition with a proof term [BoudolCastellani88] will enable the uniform derivation of expansion laws.
- Action preceded by the operators in the scope of which it occurs:

$$\theta \ ::= \ a \mid .{}_a\theta \mid \llcorner\urcorner_\rho\theta \mid \dot{+}\theta \mid +\!\!.\,\theta \mid \rfloor\!\rfloor_L\theta \mid \lfloor\!\lfloor_L\theta \mid \langle\theta,\theta\rangle_L$$

- Proved labeled transition system $(\mathsf{P}, \Theta, \longrightarrow)$ with $\longrightarrow \subseteq \mathsf{P} \times \Theta \times \mathsf{P}$.

- Since all information needed to enable reversibility is in the syntax, action prefix and choice are made static by the semantics [PU07].
- Labeling every transition with a proof term [BoudolCastellani88] will enable the uniform derivation of expansion laws.
- Action preceded by the operators in the scope of which it occurs:

$$\theta \ ::= \ a \mid ._a\theta \mid \llcorner\!\urcorner_\rho\theta \mid \dot{+}\theta \mid +\!\!\cdot\,\theta \mid \rfloor\!\rfloor_L\theta \mid \lfloor\!\lfloor_L\theta \mid \langle\theta, \theta\rangle_L$$

- Proved labeled transition system $(\mathsf{P}, \Theta, \longrightarrow)$ with $\longrightarrow \subseteq \mathsf{P} \times \Theta \times \mathsf{P}$.
- Set $\mathbb{P} \subsetneq \mathsf{P}$ of reachable processes from an initial one: $a^{\dagger} . \underline{0} \parallel_{\{a\}} \underline{0} \notin \mathbb{P}$.
- Single transition relation viewed as symmetric to meet loop property: *executed actions can be undone and undone actions can be redone*.
- Like in [DMV90] a transition $P \xrightarrow{\theta} P'$ goes:
  - forward if it is viewed as an outgoing transition of P, in which case action $act(\theta)$ is done;
  - backward if it is viewed as an incoming transition of P', in which case action $act(\theta)$ is undone.

- Operational semantic rules for action prefix (traditionally dynamic):

$$\frac{\textit{initial}(P)}{a \,.\, P \xrightarrow{a} a^{\dagger} \,.\, P} \qquad \frac{P \xrightarrow{\theta} P'}{a^{\dagger} \,.\, P \xrightarrow{\cdot a\,\theta} a^{\dagger} \,.\, P'}$$

- The prefix related to the executed action is *not discarded*.
- It becomes a †-decorated part of the target process,
  necessary to offer again that action after rolling back.
- Additional rule for performing unexecuted actions that are preceded by already executed actions (direct consequence of making prefix static).
- This second rule propagates actions executed by initial subprocesses.
- Can we view $a^{\dagger} \,.\, \_$ as the inverse operator of $a \,.\, \_$?

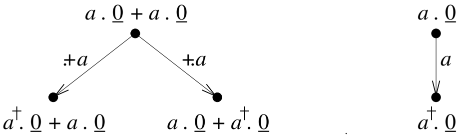
- Semantic rules for alternative composition (traditionally dynamic):

$$\frac{P_1 \xrightarrow{\theta} P_1' \quad \mathit{initial}(P_2)}{P_1 + P_2 \xrightarrow{\dot{+}\theta} P_1' + P_2} \qquad \frac{P_2 \xrightarrow{\theta} P_2' \quad \mathit{initial}(P_1)}{P_1 + P_2 \xrightarrow{+\!\!\dot{}\,\theta} P_1 + P_2'}$$

- The subprocess not involved in the executed action is *not discarded* but cannot proceed further (only the non-initial subprocess can).
- It becomes part of the target process, which is necessary for offering again the original choice after undoing all the executed actions.
- If both subprocesses are initial, both rules apply (nondet. choice).
- If not, should operator $+$ become something like $+^{\dagger}$?
  Not needed due to action decorations within either subprocess.

- The proved labeled transition system for a *sequential* process is a *tree*, whose branching points correspond to occurrences of $+$:
  - Every non-final process has at least one outgoing transition (non-final means that not all actions are decorated along one path).
  - Every non-initial process has exactly one incoming transition due to decorations associated with executed actions.

- The proved labeled transition system for a *sequential* process is a *tree*, whose branching points correspond to occurrences of $+$:
  - Every non-final process has at least one outgoing transition (non-final means that not all actions are decorated along one path).
  - Every non-initial process has exactly one incoming transition due to decorations associated with executed actions.
- Proved labeled transition systems of $a\,.\,\underline{0} + a\,.\,\underline{0}$ and $a\,.\,\underline{0}$:



- Single a-transition on the left in a forward-only process calculus.
- These two distinct processes should be considered equivalent though.

- Semantic rule for renaming (traditionally static):

$$\frac{P \xrightarrow{\theta} P'}{P \llcorner \rho \urcorner \xrightarrow{\llcorner\urcorner_\rho\, \theta} P' \llcorner \rho \urcorner}$$

- Semantic rules for parallel composition (traditionally static):

$$\frac{P_1 \xrightarrow{\theta} P_1' \quad \mathit{act}(\theta) \notin L}{P_1 \parallel_L P_2 \xrightarrow{\Vert_L \theta} P_1' \parallel_L P_2} \qquad \frac{P_2 \xrightarrow{\theta} P_2' \quad \mathit{act}(\theta) \notin L}{P_1 \parallel_L P_2 \xrightarrow{\Vert_L \theta} P_1 \parallel_L P_2'}$$

$$\frac{P_1 \xrightarrow{\theta_1} P_1' \quad P_2 \xrightarrow{\theta_2} P_2' \quad \mathit{act}(\theta_1) = \mathit{act}(\theta_2) \in L}{P_1 \parallel_L P_2 \xrightarrow{\langle \theta_1, \theta_2 \rangle_L} P_1' \parallel_L P_2'}$$

- Rule for synchronization that is sensitive to causality thanks to the presence of proof terms on transition labels:

$$\frac{P_1 \xrightarrow{\theta_1} P_1' \quad P_2 \xrightarrow{\theta_2} P_2' \quad \mathit{act}(\theta_1) = \mathit{act}(\theta_2) \in L}{P_1 \parallel_L P_2 \xrightarrow{\langle \theta_1, \theta_2 \rangle_L} \mathit{enr}(P_1' \parallel_L P_2', \langle \theta_1, \theta_2 \rangle_L)}$$

- The †-decoration of every action participating in the synchronization is enriched with a proof term of the form $\langle \theta_1, \theta_2 \rangle_L$.

- Rule for synchronization that is sensitive to causality thanks to the presence of proof terms on transition labels:

$$\frac{P_1 \xrightarrow{\theta_1} P_1' \quad P_2 \xrightarrow{\theta_2} P_2' \quad \mathit{act}(\theta_1) = \mathit{act}(\theta_2) \in L}{P_1 \parallel_L P_2 \xrightarrow{\langle \theta_1, \theta_2 \rangle_L} \mathit{enr}(P_1' \parallel_L P_2', \langle \theta_1, \theta_2 \rangle_L)}$$

- The $\dagger$-decoration of every action participating in the synchronization is enriched with a proof term of the form $\langle \theta_1, \theta_2 \rangle_L$.
- The LTS of the synchronization of autoconcurrency and autocausation $(a \, . \, \underline{0} \parallel_\emptyset a \, . \, \underline{0}) \parallel_{\{a\}} a \, . \, a \, . \, \underline{0}$ is different from the one of $a \, . \, \underline{0} \parallel_\emptyset a \, . \, \underline{0}$:

- Rule for synchronization that is sensitive to causality thanks to the presence of proof terms on transition labels:

$$\frac{P_1 \xrightarrow{\theta_1} P_1' \quad P_2 \xrightarrow{\theta_2} P_2' \quad \mathit{act}(\theta_1) = \mathit{act}(\theta_2) \in L}{P_1 \parallel_L P_2 \xrightarrow{\langle \theta_1, \theta_2 \rangle_L} \mathit{enr}(P_1' \parallel_L P_2', \langle \theta_1, \theta_2 \rangle_L)}$$

- The †-decoration of every action participating in the synchronization is enriched with a proof term of the form $\langle \theta_1, \theta_2 \rangle_L$.
- The LTS of the synchronization of autoconcurrency and autocausation $(a \,.\, \underline{0} \parallel_\emptyset a \,.\, \underline{0}) \parallel_{\{a\}} a \,.\, a \,.\, \underline{0}$ is different from the one of $a \,.\, \underline{0} \parallel_\emptyset a \,.\, \underline{0}$:
  - $(a \,.\, \underline{0} \parallel_\emptyset a \,.\, \underline{0}) \parallel_{\{a\}} a \,.\, a \,.\, \underline{0} \xrightarrow{\langle \rfloor\!\lfloor_\emptyset a, a \rangle_{\{a\}}}$

    $(a^{\dagger \langle \rfloor\!\lfloor_\emptyset a, a \rangle_{\{a\}}} \,.\, \underline{0} \parallel_\emptyset a \,.\, \underline{0}) \parallel_{\{a\}} a^{\dagger \langle \rfloor\!\lfloor_\emptyset a, a \rangle_{\{a\}}} \,.\, a \,.\, \underline{0} \xrightarrow{\langle \lfloor\!\lfloor_\emptyset a, ._a a \rangle_{\{a\}}}$

    $(a^{\dagger \langle \rfloor\!\lfloor_\emptyset a, a \rangle_{\{a\}}} \,.\, \underline{0} \parallel_\emptyset a^{\dagger \langle \lfloor\!\lfloor_\emptyset a, ._a a \rangle_{\{a\}}} \,.\, \underline{0}) \parallel_{\{a\}} a^{\dagger \langle \rfloor\!\lfloor_\emptyset a, a \rangle_{\{a\}}} \,.\, a^{\dagger \langle \lfloor\!\lfloor_\emptyset a, ._a a \rangle_{\{a\}}} \,.\, \underline{0}$

- Rule for synchronization that is sensitive to causality thanks to the presence of proof terms on transition labels:

$$\frac{P_1 \xrightarrow{\theta_1} P'_1 \quad P_2 \xrightarrow{\theta_2} P'_2 \quad \mathit{act}(\theta_1) = \mathit{act}(\theta_2) \in L}{P_1 \parallel_L P_2 \xrightarrow{\langle \theta_1, \theta_2 \rangle_L} \mathit{enr}(P'_1 \parallel_L P'_2, \langle \theta_1, \theta_2 \rangle_L)}$$

- The †-decoration of every action participating in the synchronization is enriched with a proof term of the form $\langle \theta_1, \theta_2 \rangle_L$.
- The LTS of the synchronization of autoconcurrency and autocausation $(a \,.\, \underline{0} \parallel_\emptyset a \,.\, \underline{0}) \parallel_{\{a\}} a \,.\, a \,.\, \underline{0}$ is different from the one of $a \,.\, \underline{0} \parallel_\emptyset a \,.\, \underline{0}$:
  - $(a \,.\, \underline{0} \parallel_\emptyset a \,.\, \underline{0}) \parallel_{\{a\}} a \,.\, a \,.\, \underline{0} \xrightarrow{\langle \rfloor\!\rfloor_\emptyset a, a \rangle_{\{a\}}}$
    $(a^{\dagger \langle \rfloor\!\rfloor_\emptyset a, a \rangle_{\{a\}}} \,.\, \underline{0} \parallel_\emptyset a \,.\, \underline{0}) \parallel_{\{a\}} a^{\dagger \langle \rfloor\!\rfloor_\emptyset a, a \rangle_{\{a\}}} \,.\, a \,.\, \underline{0} \xrightarrow{\langle \lfloor\!\lfloor_\emptyset a, \cdot_a a \rangle_{\{a\}}}$
    $(a^{\dagger \langle \rfloor\!\rfloor_\emptyset a, a \rangle_{\{a\}}} \,.\, \underline{0} \parallel_\emptyset a^{\dagger \langle \lfloor\!\lfloor_\emptyset a, \cdot_a a \rangle_{\{a\}}} \,.\, \underline{0}) \parallel_{\{a\}} a^{\dagger \langle \rfloor\!\rfloor_\emptyset a, a \rangle_{\{a\}}} \,.\, a^{\dagger \langle \lfloor\!\lfloor_\emptyset a, \cdot_a a \rangle_{\{a\}}} \,.\, \underline{0}$
  - $(a \,.\, \underline{0} \parallel_\emptyset a \,.\, \underline{0}) \parallel_{\{a\}} a \,.\, a \,.\, \underline{0} \xrightarrow{\langle \lfloor\!\lfloor_\emptyset a, a \rangle_{\{a\}}}$
    $(a \,.\, \underline{0} \parallel_\emptyset a^{\dagger \langle \lfloor\!\lfloor_\emptyset a, a \rangle_{\{a\}}} \,.\, \underline{0}) \parallel_{\{a\}} a^{\dagger \langle \lfloor\!\lfloor_\emptyset a, a \rangle_{\{a\}}} \,.\, a \,.\, \underline{0} \xrightarrow{\langle \rfloor\!\rfloor_\emptyset a, \cdot_a a \rangle_{\{a\}}}$
    $(a^{\dagger \langle \rfloor\!\rfloor_\emptyset a, \cdot_a a \rangle_{\{a\}}} \,.\, \underline{0} \parallel_\emptyset a^{\dagger \langle \lfloor\!\lfloor_\emptyset a, a \rangle_{\{a\}}} \,.\, \underline{0}) \parallel_{\{a\}} a^{\dagger \langle \lfloor\!\lfloor_\emptyset a, a \rangle_{\{a\}}} \,.\, a^{\dagger \langle \rfloor\!\rfloor_\emptyset a, \cdot_a a \rangle_{\{a\}}} \,.\, \underline{0}$

Strong Bisimilarities for PRPC

- Bisimulation game: *outgoing* transitions for forward direction and *incoming* transitions for backward direction [DMV90].

Strong Bisimilarities for PRPC

- Bisimulation game: *outgoing* transitions for forward direction and *incoming* transitions for backward direction [DMV90].
- A symmetric relation $\mathcal{B}$ over $\mathbb{P}$ is a:
  - Forward bisimulation iff, whenever $(P_1, P_2) \in \mathcal{B}$, then:
    - $\forall\ P_1 \xrightarrow{\theta_1} P_1'\ .\ \exists\ P_2 \xrightarrow{\theta_2} P_2'\ .\ \mathit{act}(\theta_1) = \mathit{act}(\theta_2) \wedge (P_1', P_2') \in \mathcal{B}$.

Strong Bisimilarities for PRPC

- Bisimulation game: *outgoing* transitions for forward direction and *incoming* transitions for backward direction [DMV90].
- A symmetric relation $\mathcal{B}$ over $\mathbb{P}$ is a:
  - Forward bisimulation iff, whenever $(P_1, P_2) \in \mathcal{B}$, then:
    - $\forall\ P_1 \xrightarrow{\theta_1} P_1'\ .\ \exists\ P_2 \xrightarrow{\theta_2} P_2'\ .\ \mathit{act}(\theta_1) = \mathit{act}(\theta_2) \wedge (P_1', P_2') \in \mathcal{B}$.
  - Reverse bisimulation iff, whenever $(P_1, P_2) \in \mathcal{B}$, then:
    - $\forall\ P_1' \xrightarrow{\theta_1} P_1\ .\ \exists\ P_2' \xrightarrow{\theta_2} P_2\ .\ \mathit{act}(\theta_1) = \mathit{act}(\theta_2) \wedge (P_1', P_2') \in \mathcal{B}$.

Strong Bisimilarities for PRPC

- Bisimulation game: *outgoing* transitions for forward direction and *incoming* transitions for backward direction [DMV90].
- A symmetric relation $\mathcal{B}$ over $\mathbb{P}$ is a:
  - Forward bisimulation iff, whenever $(P_1, P_2) \in \mathcal{B}$, then:
    - $\forall\ P_1 \xrightarrow{\theta_1} P_1'\ .\ \exists\ P_2 \xrightarrow{\theta_2} P_2'\ .\ \mathit{act}(\theta_1) = \mathit{act}(\theta_2) \wedge (P_1', P_2') \in \mathcal{B}$.
  - Reverse bisimulation iff, whenever $(P_1, P_2) \in \mathcal{B}$, then:
    - $\forall\ P_1' \xrightarrow{\theta_1} P_1\ .\ \exists\ P_2' \xrightarrow{\theta_2} P_2\ .\ \mathit{act}(\theta_1) = \mathit{act}(\theta_2) \wedge (P_1', P_2') \in \mathcal{B}$.
  - Forward-reverse bisimulation iff, whenever $(P_1, P_2) \in \mathcal{B}$, then:
    - $\forall\ P_1 \xrightarrow{\theta_1} P_1'\ .\ \exists\ P_2 \xrightarrow{\theta_2} P_2'\ .\ \mathit{act}(\theta_1) = \mathit{act}(\theta_2) \wedge (P_1', P_2') \in \mathcal{B}$.
    - $\forall\ P_1' \xrightarrow{\theta_1} P_1\ .\ \exists\ P_2' \xrightarrow{\theta_2} P_2\ .\ \mathit{act}(\theta_1) = \mathit{act}(\theta_2) \wedge (P_1', P_2') \in \mathcal{B}$.

Strong Bisimilarities for PRPC

- Bisimulation game: *outgoing* transitions for forward direction and *incoming* transitions for backward direction [DMV90].
- A symmetric relation $\mathcal{B}$ over $\mathbb{P}$ is a:
  - Forward bisimulation iff, whenever $(P_1, P_2) \in \mathcal{B}$, then:
    - $\forall\, P_1 \xrightarrow{\theta_1} P_1' \,.\, \exists\, P_2 \xrightarrow{\theta_2} P_2' \,.\, \mathit{act}(\theta_1) = \mathit{act}(\theta_2) \wedge (P_1', P_2') \in \mathcal{B}$.
  - Reverse bisimulation iff, whenever $(P_1, P_2) \in \mathcal{B}$, then:
    - $\forall\, P_1' \xrightarrow{\theta_1} P_1 \,.\, \exists\, P_2' \xrightarrow{\theta_2} P_2 \,.\, \mathit{act}(\theta_1) = \mathit{act}(\theta_2) \wedge (P_1', P_2') \in \mathcal{B}$.
  - Forward-reverse bisimulation iff, whenever $(P_1, P_2) \in \mathcal{B}$, then:
    - $\forall\, P_1 \xrightarrow{\theta_1} P_1' \,.\, \exists\, P_2 \xrightarrow{\theta_2} P_2' \,.\, \mathit{act}(\theta_1) = \mathit{act}(\theta_2) \wedge (P_1', P_2') \in \mathcal{B}$.
    - $\forall\, P_1' \xrightarrow{\theta_1} P_1 \,.\, \exists\, P_2' \xrightarrow{\theta_2} P_2 \,.\, \mathit{act}(\theta_1) = \mathit{act}(\theta_2) \wedge (P_1', P_2') \in \mathcal{B}$.
- Largest such relations: $\sim_{\mathrm{FB}}$, $\sim_{\mathrm{RB}}$, $\sim_{\mathrm{FRB}}$.
- In order for $P_1, P_2 \in \mathbb{P}$ to be identified by $\sim_{\mathrm{FB}} / \sim_{\mathrm{RB}}$ their forward/backward ready sets must coincide.

Discriminating Power

- $\sim_{\rm FRB} \subsetneq \sim_{\rm FB} \cap \sim_{\rm RB}$:
    - The inclusion is strict because the two processes $a^{\dagger} . \underline{0}$ and $a^{\dagger} . \underline{0} + c . \underline{0}$ are identified by $\sim_{\rm FB}$ and $\sim_{\rm RB}$, but distinguished by $\sim_{\rm FRB}$.
    - $\sim_{\rm FB}$ and $\sim_{\rm RB}$ are incomparable because $a^{\dagger} . \underline{0} \sim_{\rm FB} \underline{0}$ but $a^{\dagger} . \underline{0} \not\sim_{\rm RB} \underline{0}$ while $a . \underline{0} \sim_{\rm RB} \underline{0}$ but $a . \underline{0} \not\sim_{\rm FB} \underline{0}$.

Discriminating Power

- $\sim_{\rm FRB} \subsetneq \sim_{\rm FB} \cap \sim_{\rm RB}$:
  - The inclusion is strict because the two processes $a^{\dagger} \, . \, \underline{0}$ and $a^{\dagger} \, . \, \underline{0} + c \, . \, \underline{0}$ are identified by $\sim_{\rm FB}$ and $\sim_{\rm RB}$, but distinguished by $\sim_{\rm FRB}$.
  - $\sim_{\rm FB}$ and $\sim_{\rm RB}$ are incomparable because $a^{\dagger} \, . \, \underline{0} \sim_{\rm FB} \underline{0}$ but $a^{\dagger} \, . \, \underline{0} \not\sim_{\rm RB} \underline{0}$ while $a \, . \, \underline{0} \sim_{\rm RB} \underline{0}$ but $a \, . \, \underline{0} \not\sim_{\rm FB} \underline{0}$.
- First comparative remark ($\sim_{\rm FB}$ vs. $\sim_{\rm RB}$):
  - $\sim_{\rm FRB} = \sim_{\rm FB}$ over initial processes, with $\sim_{\rm RB}$ strictly coarser.
  - $\sim_{\rm FRB} \neq \sim_{\rm RB}$ over final processes because, after going backward, discarded subprocesses come into play again for $\sim_{\rm FRB}$.

Discriminating Power

- $\sim_{\rm FRB} \subsetneq \sim_{\rm FB} \cap \sim_{\rm RB}$:
  - The inclusion is strict because the two processes $a^{\dagger} . \underline{0}$ and $a^{\dagger} . \underline{0} + c . \underline{0}$ are identified by $\sim_{\rm FB}$ and $\sim_{\rm RB}$, but distinguished by $\sim_{\rm FRB}$.
  - $\sim_{\rm FB}$ and $\sim_{\rm RB}$ are incomparable because $a^{\dagger} . \underline{0} \sim_{\rm FB} \underline{0}$ but $a^{\dagger} . \underline{0} \not\sim_{\rm RB} \underline{0}$ while $a . \underline{0} \sim_{\rm RB} \underline{0}$ but $a . \underline{0} \not\sim_{\rm FB} \underline{0}$.
- First comparative remark ($\sim_{\rm FB}$ vs. $\sim_{\rm RB}$):
  - $\sim_{\rm FRB} = \sim_{\rm FB}$ over initial processes, with $\sim_{\rm RB}$ strictly coarser.
  - $\sim_{\rm FRB} \neq \sim_{\rm RB}$ over final processes because, after going backward, discarded subprocesses come into play again for $\sim_{\rm FRB}$.
- $a . \underline{0} + a . \underline{0}$ and $a . \underline{0}$ are identified by all three bisimilarities as witnessed by any bisimulation containing the pairs $(a . \underline{0} + a . \underline{0}, a . \underline{0})$, $(a^{\dagger} . \underline{0} + a . \underline{0}, a^{\dagger} . \underline{0})$, $(a . \underline{0} + a^{\dagger} . \underline{0}, a^{\dagger} . \underline{0})$.

Compositionality Properties

- $\sim_{\mathrm{FB}}$ equates processes with different past: $a_1^{\dagger} \,.\, \underline{0} \sim_{\mathrm{FB}} a_2^{\dagger} \,.\, \underline{0} \sim_{\mathrm{FB}} \underline{0}$.
- $\sim_{\mathrm{RB}}$ equates processes with different future: $a_1 \,.\, \underline{0} \sim_{\mathrm{RB}} a_2 \,.\, \underline{0} \sim_{\mathrm{RB}} \underline{0}$.

Compositionality Properties

- $\sim_{\text{FB}}$ equates processes with different past: $a_1^\dagger \, . \, \underline{0} \sim_{\text{FB}} a_2^\dagger \, . \, \underline{0} \sim_{\text{FB}} \underline{0}$.
- $\sim_{\text{RB}}$ equates processes with different future: $a_1 \, . \, \underline{0} \sim_{\text{RB}} a_2 \, . \, \underline{0} \sim_{\text{RB}} \underline{0}$.
- Second comparative remark ($\sim_{\text{FB}}$ vs. $\sim_{\text{RB}}$):
  - $a^\dagger . b . \underline{0} \sim_{\text{FB}} b . \underline{0}$ but $a^\dagger . b . \underline{0} + c . \underline{0} \not\sim_{\text{FB}} b . \underline{0} + c . \underline{0}$.
  - $a^\dagger . b . \underline{0} \not\sim_{\text{RB}} b . \underline{0}$ hence no such compositionality violation for $\sim_{\text{RB}}$.

Compositionality Properties

- $\sim_{\rm FB}$ equates processes with different past: $a_1^{\dagger} \,.\, \underline{0} \sim_{\rm FB} a_2^{\dagger} \,.\, \underline{0} \sim_{\rm FB} \underline{0}$.
- $\sim_{\rm RB}$ equates processes with different future: $a_1 \,.\, \underline{0} \sim_{\rm RB} a_2 \,.\, \underline{0} \sim_{\rm RB} \underline{0}$.
- Second comparative remark ($\sim_{\rm FB}$ vs. $\sim_{\rm RB}$):
  - $a^{\dagger} \,.\, b \,.\, \underline{0} \sim_{\rm FB} b \,.\, \underline{0}$ but $a^{\dagger} \,.\, b \,.\, \underline{0} + c \,.\, \underline{0} \not\sim_{\rm FB} b \,.\, \underline{0} + c \,.\, \underline{0}$.
  - $a^{\dagger} \,.\, b \,.\, \underline{0} \not\sim_{\rm RB} b \,.\, \underline{0}$ hence no such compositionality violation for $\sim_{\rm RB}$.
- $\sim_{\rm RB}$ and $\sim_{\rm FRB}$ never identify an initial process with a non-initial one, hence $\sim_{\rm FB}$ has to be made sensitive to the *presence of the past*.
- A symmetric relation $\mathcal{B}$ over $\mathbb{P}$ is a past-sensitive forward bisimulation iff it is a forward bisimulation in which *initial*$(P_1) \Longleftrightarrow$ *initial*(P_2) for all $(P_1, P_2) \in \mathcal{B}$.
- Largest such relation: $\sim_{\rm FB:ps}$.
- $a_1^{\dagger} \,.\, \underline{0} \sim_{\rm FB:ps} a_2^{\dagger} \,.\, \underline{0}$, but $a^{\dagger} \,.\, \underline{0} \not\sim_{\rm FB:ps} \underline{0}$ and $a^{\dagger} \,.\, b \,.\, \underline{0} \not\sim_{\rm FB:ps} b \,.\, \underline{0}$.

- Let $P_1, P_2 \in \mathbb{P}$ be such that $P_1 \sim P_2$ and take arbitrary a, ρ, L, P.
- All strong bisimilarities are congruences w.r.t. action prefix:
  - $a \,.\, P_1 \sim a \,.\, P_2$ provided that *initial*$(P_1) \wedge$ *initial*(P_2).
  - $a^\dagger . \, P_1 \sim a^\dagger . \, P_2$.
- All strong bisimilarities are congruences w.r.t. renaming:
  - $P_1 \llcorner \rho \urcorner \sim P_2 \llcorner \rho \urcorner$.
- All strong bisimilarities are congruences w.r.t. parallel composition:
  - $P_1 \parallel_L P \sim P_2 \parallel_L P$ and $P \parallel_L P_1 \sim P \parallel_L P_2$ provided that $P_1 \parallel_L P, P_2 \parallel_L P, P \parallel_L P_1, P \parallel_L P_2 \in \mathbb{P}$.

- Let $P_1, P_2 \in \mathbb{P}$ be such that $P_1 \sim P_2$ and take arbitrary a, ρ, L, P.
- All strong bisimilarities are congruences w.r.t. action prefix:
  - $a \,.\, P_1 \sim a \,.\, P_2$ provided that *initial*$(P_1) \wedge$ *initial*(P_2).
  - $a^\dagger .\, P_1 \sim a^\dagger .\, P_2$.
- All strong bisimilarities are congruences w.r.t. renaming:
  - $P_1 \llcorner \rho \urcorner \sim P_2 \llcorner \rho \urcorner$.
- All strong bisimilarities are congruences w.r.t. parallel composition:
  - $P_1 \parallel_L P \sim P_2 \parallel_L P$ and $P \parallel_L P_1 \sim P \parallel_L P_2$ provided that $P_1 \parallel_L P, P_2 \parallel_L P, P \parallel_L P_1, P \parallel_L P_2 \in \mathbb{P}$.
- $\sim_{\mathrm{FB:ps}}$, $\sim_{\mathrm{RB}}$, $\sim_{\mathrm{FRB}}$ are congruences w.r.t. alternative composition:
  - $P_1 + P \sim P_2 + P$ and $P + P_1 \sim P + P_2$ provided that *initial*$(P) \vee ($*initial*$(P_1) \wedge$ *initial*$(P_2))$.
- $\sim_{\mathrm{FB:ps}}$ is the coarsest congruence w.r.t. $+$ contained in $\sim_{\mathrm{FB}}$:
  - $P_1 \sim_{\mathrm{FB:ps}} P_2$ iff $P_1 + P \sim_{\mathrm{FB}} P_2 + P$ for all $P \in \mathbb{P}$ s.t. *initial*$(P) \vee ($*initial*$(P_1) \wedge$ *initial*$(P_2))$.

Modal Logic Characterizations

- Properties preserved by each equivalence; diagnostic information via distinguishing formulas explaining why two processes are not bisimilar.
- Hennessy-Milner logic extended with a backward modality (and init) from which suitable fragments are taken.
- Syntax:

$$\phi \ ::= \ \mathsf{true} \mid \mathsf{init} \mid \neg\phi \mid \phi \wedge \phi \mid \langle a \rangle \phi \mid \langle a^{\dagger} \rangle \phi$$

Modal Logic Characterizations

- Properties preserved by each equivalence; diagnostic information via distinguishing formulas explaining why two processes are not bisimilar.
- Hennessy-Milner logic extended with a backward modality (and init) from which suitable fragments are taken.
- Syntax:

$$\phi \ ::= \ \mathsf{true} \mid \mathsf{init} \mid \neg\phi \mid \phi \wedge \phi \mid \langle a \rangle \phi \mid \langle a^{\dagger} \rangle \phi$$

- Semantics:

$$\begin{array}{lcll}
P & \models & \mathsf{true} & \text{for all } P \in \mathbb{P} \\
P & \models & \mathsf{init} & \text{iff } \mathit{initial}(P) \\
P & \models & \neg\phi & \text{iff } P \not\models \phi \\
P & \models & \phi_1 \wedge \phi_2 & \text{iff } P \models \phi_1 \text{ and } P \models \phi_2 \\
P & \models & \langle a \rangle \phi & \text{iff there exists } P \xrightarrow{\theta} P' \text{ s.t. } \mathit{act}(\theta) = a \text{ and } P' \models \phi \\
P & \models & \langle a^{\dagger} \rangle \phi & \text{iff there exists } P' \xrightarrow{\theta} P \text{ s.t. } \mathit{act}(\theta) = a \text{ and } P' \models \phi
\end{array}$$

- Fragments characterizing the four strong bisimilarities:

	true	init	$\neg$	$\wedge$	$\langle a\rangle$	$\langle a^\dagger\rangle$
$\mathcal{L}_{\mathrm{FB}}$	✓		✓	✓	✓	
$\mathcal{L}_{\mathrm{FB:ps}}$	✓	✓	✓	✓	✓	
$\mathcal{L}_{\mathrm{RB}}$	✓					✓
$\mathcal{L}_{\mathrm{FRB}}$	✓		✓	✓	✓	✓

- $\mathcal{L}_{\mathrm{FB}}$ / $\mathcal{L}_{\mathrm{FB:ps}}$ / $\mathcal{L}_{\mathrm{RB}}$ / $\mathcal{L}_{\mathrm{FRB}}$ characterizes $\sim_{\mathrm{FB}}$ / $\sim_{\mathrm{FB:ps}}$ / $\sim_{\mathrm{RB}}$ / $\sim_{\mathrm{FRB}}$:

  $$P_1 \sim_B P_2 \text{ iff } \forall \phi \in \mathcal{L}_B.\, P_1 \models \phi \iff P_2 \models \phi$$

- Fragments characterizing the four strong bisimilarities:

	true	init	$\neg$	$\wedge$	$\langle a \rangle$	$\langle a^{\dagger} \rangle$
$\mathcal{L}_{\rm FB}$	$\checkmark$		$\checkmark$	$\checkmark$	$\checkmark$	
$\mathcal{L}_{\rm FB:ps}$	$\checkmark$	$\checkmark$	$\checkmark$	$\checkmark$	$\checkmark$	
$\mathcal{L}_{\rm RB}$	$\checkmark$					$\checkmark$
$\mathcal{L}_{\rm FRB}$	$\checkmark$		$\checkmark$	$\checkmark$	$\checkmark$	$\checkmark$

- $\mathcal{L}_{\rm FB}$ / $\mathcal{L}_{\rm FB:ps}$ / $\mathcal{L}_{\rm RB}$ / $\mathcal{L}_{\rm FRB}$ characterizes $\sim_{\rm FB}$ / $\sim_{\rm FB:ps}$ / $\sim_{\rm RB}$ / $\sim_{\rm FRB}$:
$$P_1 \sim_B P_2 \text{ iff } \forall \phi \in \mathcal{L}_B .\, P_1 \models \phi \iff P_2 \models \phi$$
- $\sim_{\rm RB}$ boils down to reverse trace equivalence!
- Obvious over sequential processes because each of them has at most one incoming transition due to executed actions being decorated.

Equational Characterizations

- Fundamental equational laws; exploitable as bisimilarity-preserving rewriting rules for manipulating processes.
- Deduction system $\vdash$ based on these axioms and inference rules due to $\sim_{\mathrm{FB:ps}}$, $\sim_{\mathrm{RB}}$, $\sim_{\mathrm{FRB}}$ being equivalence relations and congruences:

Equational Characterizations

- Fundamental equational laws; exploitable as bisimilarity-preserving rewriting rules for manipulating processes.
- Deduction system $\vdash$ based on these axioms and inference rules due to $\sim_{\mathrm{FB:ps}}$, $\sim_{\mathrm{RB}}$, $\sim_{\mathrm{FRB}}$ being equivalence relations and congruences:
  - Reflexivity $P = P$, symmetry $\dfrac{P_1 = P_2}{P_2 = P_1}$, transitivity $\dfrac{P_1 = P_2 \quad P_2 = P_3}{P_1 = P_3}$.
  - .-Substitutivity: $\dfrac{P_1 = P_2 \quad \mathit{initial}(P_1) \wedge \mathit{initial}(P_2)}{a \,.\, P_1 = a \,.\, P_2}$, $\dfrac{P_1 = P_2}{a^{\dagger} \,.\, P_1 = a^{\dagger} \,.\, P_2}$.
  - $\llcorner\urcorner$-substitutivity: $\dfrac{P_1 = P_2}{P_1 \llcorner\rho\urcorner = P_2 \llcorner\rho\urcorner}$.
  - $+$-Substitutivity: $\dfrac{P_1 = P_2 \quad \mathit{initial}(P) \vee (\mathit{initial}(P_1) \wedge \mathit{initial}(P_2))}{P_1 + P = P_2 + P \quad P + P_1 = P + P_2}$.
  - $\|$-substitutivity: $\dfrac{P_1 = P_2 \quad P_1 \,\|_L\, P, P_2 \,\|_L\, P, P \,\|_L\, P_1, P \,\|_L\, P_2 \in \mathbb{P}}{P_1 \,\|_L\, P = P_2 \,\|_L\, P \quad P \,\|_L\, P_1 = P \,\|_L\, P_2}$.
- $\vdash$ is sound and complete w.r.t. $\sim$ when $\vdash P_1 = P_2$ iff $P_1 \sim P_2$.

- Operator-specific axioms for renaming-free sequential processes:

(A_1)		$(P + Q) + R$	$=$	$P + (Q + R)$	where at least two are initial
(A_2)		$P + Q$	$=$	$Q + P$	where $\mathit{initial}(P) \vee \mathit{initial}(Q)$
(A_3)		$P + \underline{0}$	$=$	P	
(A_4)	$[\sim_{\mathrm{FB:ps}}]$	$a^{\dagger} . P$	$=$	$b^{\dagger} . P$	if $\mathit{initial}(P)$
(A_5)	$[\sim_{\mathrm{FB:ps}}]$	$a^{\dagger} . P$	$=$	P	if $\neg \mathit{initial}(P)$
(A_6)	$[\sim_{\mathrm{FB:ps}}]$	$P + Q$	$=$	P	if $\neg \mathit{initial}(P)$, where $\mathit{initial}(Q)$
(A_7)	$[\sim_{\mathrm{RB}}]$	$a . P$	$=$	P	where $\mathit{initial}(P)$
(A_8)	$[\sim_{\mathrm{RB}}]$	$P + Q$	$=$	P	if $\mathit{initial}(Q)$
(A_9)	$[\sim_{\mathrm{FB:ps}}]$	$P + P$	$=$	P	where $\mathit{initial}(P)$
(A_{10})	$[\sim_{\mathrm{FRB}}]$	$P + Q$	$=$	P	if $\mathit{initial}(Q) \wedge \mathit{to\_initial}(P) = Q$

- A_8 subsumes A_3 (with $Q = \underline{0}$) and A_9 (with $Q = P$).
- A_9 and A_6 apply in two different cases (P initial or not).
- A_{10} originally developed in [LanesePhillips21].
- $\vdash^{1,2,3}_{4,5,6,9}$ / $\vdash^{1,2}_{7,8}$ / $\vdash^{1,2,3}_{10}$ sound and complete for $\sim_{\mathrm{FB:ps}}$ / $\sim_{\mathrm{RB}}$ / $\sim_{\mathrm{FRB}}$.
- Third comparative remark: explicit vs. implicit idempotency.

- Axioms for renaming:

(A_{11})	$\underline{0}\llcorner\rho\urcorner$	$=$	$\underline{0}$	
(A_{12})	$(a\,.\,P)\llcorner\rho\urcorner$	$=$	$\rho(a)\,.\,(P\llcorner\rho\urcorner)$	where $\mathit{initial}(P)$
(A_{13})	$(a^{\dagger}.\,P)\llcorner\rho\urcorner$	$=$	$\rho(a)^{\dagger}.\,(P\llcorner\rho\urcorner)$	
(A_{14})	$(P+Q)\llcorner\rho\urcorner$	$=$	$(P\llcorner\rho\urcorner)+(Q\llcorner\rho\urcorner)$	where $\mathit{initial}(P)\vee\mathit{initial}(Q)$

- They progressively remove all occurrences of renaming.
- $\sim_{\mathrm{FB:ps}}$ needs all of them.
- $\sim_{\mathrm{RB}}$ only needs A_{11} and A_{13}.
- $\sim_{\mathrm{FRB}}$ needs all of them.
- We will see later on expansion laws for parallel composition.

Weak Bisimilarities for PRPC

- Abstracting from possibly empty sequences $\Longrightarrow$ of τ-transitions: $\stackrel{\hat{\theta}}{\Longrightarrow} \; = \; \Longrightarrow$ if $act(\theta) = \tau$, $\stackrel{\hat{\theta}}{\Longrightarrow} \; = \; \Longrightarrow \stackrel{\theta}{\longrightarrow} \Longrightarrow$ if $act(\theta) \neq \tau$.

Weak Bisimilarities for PRPC

- Abstracting from possibly empty sequences $\Longrightarrow$ of τ-transitions: $\stackrel{\hat{\theta}}{\Longrightarrow} \; = \; \Longrightarrow$ if $act(\theta) = \tau$, $\stackrel{\hat{\theta}}{\Longrightarrow} \; = \; \Longrightarrow \stackrel{\theta}{\longrightarrow} \Longrightarrow$ if $act(\theta) \neq \tau$.
- A symmetric relation $\mathcal{B}$ over $\mathbb{P}$ is a:
  - Weak forward bisimulation iff, whenever $(P_1, P_2) \in \mathcal{B}$, then:
    - $\forall \; P_1 \xrightarrow{\theta_1} P_1' \; . \; \exists \; P_2 \stackrel{\hat{\theta}_2}{\Longrightarrow} P_2' \; . \; act(\theta_1) = act(\theta_2) \wedge (P_1', P_2') \in \mathcal{B}$.

Weak Bisimilarities for PRPC

- Abstracting from possibly empty sequences $\Longrightarrow$ of τ-transitions: $\stackrel{\hat{\theta}}{\Longrightarrow} \; = \; \Longrightarrow$ if $act(\theta) = \tau$, $\stackrel{\hat{\theta}}{\Longrightarrow} \; = \; \Longrightarrow \stackrel{\theta}{\longrightarrow} \Longrightarrow$ if $act(\theta) \neq \tau$.
- A symmetric relation $\mathcal{B}$ over $\mathbb{P}$ is a:
  - Weak forward bisimulation iff, whenever $(P_1, P_2) \in \mathcal{B}$, then:
    - $\forall \; P_1 \stackrel{\theta_1}{\longrightarrow} P_1' \; . \; \exists \; P_2 \stackrel{\hat{\theta}_2}{\Longrightarrow} P_2' \; . \; act(\theta_1) = act(\theta_2) \wedge (P_1', P_2') \in \mathcal{B}$.
  - Weak reverse bisimulation iff, whenever $(P_1, P_2) \in \mathcal{B}$, then:
    - $\forall \; P_1' \stackrel{\theta_1}{\longrightarrow} P_1 \; . \; \exists \; P_2' \stackrel{\hat{\theta}_2}{\Longrightarrow} P_2 \; . \; act(\theta_1) = act(\theta_2) \wedge (P_1', P_2') \in \mathcal{B}$.

Weak Bisimilarities for PRPC

- Abstracting from possibly empty sequences $\Longrightarrow$ of τ-transitions:
  $\stackrel{\hat{\theta}}{\Longrightarrow} \; = \; \Longrightarrow$ if $act(\theta) = \tau$, $\stackrel{\hat{\theta}}{\Longrightarrow} \; = \; \Longrightarrow \stackrel{\theta}{\longrightarrow} \Longrightarrow$ if $act(\theta) \neq \tau$.
- A symmetric relation $\mathcal{B}$ over $\mathbb{P}$ is a:
  - Weak forward bisimulation iff, whenever $(P_1, P_2) \in \mathcal{B}$, then:
    - $\forall \; P_1 \xrightarrow{\theta_1} P_1' \; . \; \exists \; P_2 \stackrel{\hat{\theta}_2}{\Longrightarrow} P_2' \; . \; act(\theta_1) = act(\theta_2) \wedge (P_1', P_2') \in \mathcal{B}$.
  - Weak reverse bisimulation iff, whenever $(P_1, P_2) \in \mathcal{B}$, then:
    - $\forall \; P_1' \xrightarrow{\theta_1} P_1 \; . \; \exists \; P_2' \stackrel{\hat{\theta}_2}{\Longrightarrow} P_2 \; . \; act(\theta_1) = act(\theta_2) \wedge (P_1', P_2') \in \mathcal{B}$.
  - Weak forward-reverse bisimulation iff, whenever $(P_1, P_2) \in \mathcal{B}$, then:
    - $\forall \; P_1 \xrightarrow{\theta_1} P_1' \; . \; \exists \; P_2 \stackrel{\hat{\theta}_2}{\Longrightarrow} P_2' \; . \; act(\theta_1) = act(\theta_2) \wedge (P_1', P_2') \in \mathcal{B}$.
    - $\forall \; P_1' \xrightarrow{\theta_1} P_1 \; . \; \exists \; P_2' \stackrel{\hat{\theta}_2}{\Longrightarrow} P_2 \; . \; act(\theta_1) = act(\theta_2) \wedge (P_1', P_2') \in \mathcal{B}$.

Weak Bisimilarities for PRPC

- Abstracting from possibly empty sequences $\Longrightarrow$ of τ-transitions:
  $\stackrel{\hat{\theta}}{\Longrightarrow} \; = \; \Longrightarrow$ if $act(\theta) = \tau$, $\stackrel{\hat{\theta}}{\Longrightarrow} \; = \; \Longrightarrow \stackrel{\theta}{\longrightarrow} \Longrightarrow$ if $act(\theta) \neq \tau$.
- A symmetric relation $\mathcal{B}$ over $\mathbb{P}$ is a:
  - Weak forward bisimulation iff, whenever $(P_1, P_2) \in \mathcal{B}$, then:
    - $\forall \; P_1 \stackrel{\theta_1}{\longrightarrow} P_1' \; . \; \exists \; P_2 \stackrel{\hat{\theta}_2}{\Longrightarrow} P_2' \; . \; act(\theta_1) = act(\theta_2) \wedge (P_1', P_2') \in \mathcal{B}$.
  - Weak reverse bisimulation iff, whenever $(P_1, P_2) \in \mathcal{B}$, then:
    - $\forall \; P_1' \stackrel{\theta_1}{\longrightarrow} P_1 \; . \; \exists \; P_2' \stackrel{\hat{\theta}_2}{\Longrightarrow} P_2 \; . \; act(\theta_1) = act(\theta_2) \wedge (P_1', P_2') \in \mathcal{B}$.
  - Weak forward-reverse bisimulation iff, whenever $(P_1, P_2) \in \mathcal{B}$, then:
    - $\forall \; P_1 \stackrel{\theta_1}{\longrightarrow} P_1' \; . \; \exists \; P_2 \stackrel{\hat{\theta}_2}{\Longrightarrow} P_2' \; . \; act(\theta_1) = act(\theta_2) \wedge (P_1', P_2') \in \mathcal{B}$.
    - $\forall \; P_1' \stackrel{\theta_1}{\longrightarrow} P_1 \; . \; \exists \; P_2' \stackrel{\hat{\theta}_2}{\Longrightarrow} P_2 \; . \; act(\theta_1) = act(\theta_2) \wedge (P_1', P_2') \in \mathcal{B}$.
- Largest such relations: $\approx_{\mathrm{FB}}$, $\approx_{\mathrm{RB}}$, $\approx_{\mathrm{FRB}}$.
- Alternative definitions with $\stackrel{\hat{\theta}_1}{\Longrightarrow}$ in place of $\stackrel{\theta_1}{\longrightarrow}$.
- In order for $P_1, P_2 \in \mathbb{P}$ to be identified by $\approx_{\mathrm{FB}} / \approx_{\mathrm{RB}}$ their weak forward/backward ready sets have to coincide.

Discriminating Power

- Each weak bisimilarity is strictly coarser than its strong counterpart.
- $\approx_{\mathrm{FRB}} \subsetneq \approx_{\mathrm{FB}} \cap \approx_{\mathrm{RB}}$ with $\approx_{\mathrm{FB}}$ and $\approx_{\mathrm{RB}}$ being incomparable.

Discriminating Power

- Each weak bisimilarity is strictly coarser than its strong counterpart.
- $\approx_{\mathrm{FRB}} \subsetneq \approx_{\mathrm{FB}} \cap \approx_{\mathrm{RB}}$ with $\approx_{\mathrm{FB}}$ and $\approx_{\mathrm{RB}}$ being incomparable.
- $\approx_{\mathrm{FRB}} \neq \approx_{\mathrm{FB}}$ over initial processes:
  - $\tau \,.\, a \,.\, \underline{0} + a \,.\, \underline{0} + b \,.\, \underline{0}$ and $\tau \,.\, a \,.\, \underline{0} + b \,.\, \underline{0}$ are identified by $\approx_{\mathrm{FB}}$ but told apart by $\approx_{\mathrm{FRB}}$
    - Doing a on the left is matched by doing τ and then a on the right.
    - Undoing a on the right cannot be matched on the left.
  - $c \,.\, (\tau \,.\, a \,.\, \underline{0} + a \,.\, \underline{0} + b \,.\, \underline{0})$ and $c \,.\, (\tau \,.\, a \,.\, \underline{0} + b \,.\, \underline{0})$ is an analogous counterexample with non-initial τ-actions:
    - Doing c on one side is matched by doing c on the other side.
    - Doing a on the left is matched by doing τ and then a on the right.
    - Undoing a on the right cannot be matched on the left.

Compositionality Properties

- Neither $\approx_{\mathrm{FB}}$ nor $\approx_{\mathrm{FRB}}$ is compositional:
  - $a^{\dagger} . b . \underline{0} \approx_{\mathrm{FB}} b . \underline{0}$ but $a^{\dagger} . b . \underline{0} + c . \underline{0} \not\approx_{\mathrm{FB}} b . \underline{0} + c . \underline{0}$ (same as $\sim_{\mathrm{FB}}$).
  - $\tau . a . \underline{0} \approx_{\mathrm{FB}} a . \underline{0}$ but $\tau . a . \underline{0} + b . \underline{0} \not\approx_{\mathrm{FB}} a . \underline{0} + b . \underline{0}$.
  - $\tau . a . \underline{0} \approx_{\mathrm{FRB}} a . \underline{0}$ but $\tau . a . \underline{0} + b . \underline{0} \not\approx_{\mathrm{FRB}} a . \underline{0} + b . \underline{0}$.
- The weak congruence construction à la Milner does not work here, past sensitivity is the solution again.

Compositionality Properties

- Neither $\approx_{\rm FB}$ nor $\approx_{\rm FRB}$ is compositional:
  - $a^{\dagger} . b . \underline{0} \approx_{\rm FB} b . \underline{0}$ but $a^{\dagger} . b . \underline{0} + c . \underline{0} \not\approx_{\rm FB} b . \underline{0} + c . \underline{0}$ (same as $\sim_{\rm FB}$).
  - $\tau . a . \underline{0} \approx_{\rm FB} a . \underline{0}$ but $\tau . a . \underline{0} + b . \underline{0} \not\approx_{\rm FB} a . \underline{0} + b . \underline{0}$.
  - $\tau . a . \underline{0} \approx_{\rm FRB} a . \underline{0}$ but $\tau . a . \underline{0} + b . \underline{0} \not\approx_{\rm FRB} a . \underline{0} + b . \underline{0}$.
- The weak congruence construction à la Milner does not work here, past sensitivity is the solution again.
- A symmetric relation $\mathcal{B}$ over $\mathbb{P}$ is a weak past-sensitive forward bisim. iff it is a weak forward bisim. in which *initial*$(P_1) \iff$ *initial*(P_2) for all $(P_1, P_2) \in \mathcal{B}$.
- A symm. rel. $\mathcal{B}$ over $\mathbb{P}$ is a weak past-sensitive forward-reverse bisim. iff it is a weak forward-reverse bisim. s.t. *initial*$(P_1) \iff$ *initial*(P_2) for all $(P_1, P_2) \in \mathcal{B}$.
- Largest such relations: $\approx_{\rm FB:ps}$, $\approx_{\rm FRB:ps}$.
- $\sim_{\rm FRB} \subsetneq \approx_{\rm FRB:ps}$ as the former satisfies the initiality condition.

- Let $P_1, P_2 \in \mathbb{P}$ be such that $P_1 \approx P_2$ and take arbitrary a, ρ, L, P.
- All weak bisimilarities are congruences w.r.t. action prefix:
  - $a \, . \, P_1 \approx a \, . \, P_2$ provided that *initial*$(P_1) \wedge$ *initial*(P_2).
  - $a^\dagger . \, P_1 \approx a^\dagger . \, P_2$.
- All weak bisimilarities are congruences w.r.t. renaming:
  - $P_1 \llcorner \rho \urcorner \approx P_2 \llcorner \rho \urcorner$.
- All weak bisimilarities are congruences w.r.t. parallel composition:
  - $P_1 \parallel_L P \approx P_2 \parallel_L P$ and $P \parallel_L P_1 \approx P \parallel_L P_2$ provided that $P_1 \parallel_L P, P_2 \parallel_L P, P \parallel_L P_1, P \parallel_L P_2 \in \mathbb{P}$.

- Let $P_1, P_2 \in \mathbb{P}$ be such that $P_1 \approx P_2$ and take arbitrary a, ρ, L, P.
- All weak bisimilarities are congruences w.r.t. action prefix:
  - $a \,.\, P_1 \approx a \,.\, P_2$ provided that *initial*$(P_1) \wedge$ *initial*(P_2).
  - $a^\dagger \,.\, P_1 \approx a^\dagger \,.\, P_2$.
- All weak bisimilarities are congruences w.r.t. renaming:
  - $P_1 \llcorner \rho \urcorner \approx P_2 \llcorner \rho \urcorner$.
- All weak bisimilarities are congruences w.r.t. parallel composition:
  - $P_1 \parallel_L P \approx P_2 \parallel_L P$ and $P \parallel_L P_1 \approx P \parallel_L P_2$
    provided that $P_1 \parallel_L P, P_2 \parallel_L P, P \parallel_L P_1, P \parallel_L P_2 \in \mathbb{P}$.
- $\approx_{\mathrm{FB:ps}}$, $\approx_{\mathrm{RB}}$, $\approx_{\mathrm{FRB:ps}}$ are congruences w.r.t. alternative composition:
  - $P_1 + P \approx P_2 + P$ and $P + P_1 \approx P + P_2$
    provided that *initial*$(P) \vee ($*initial*$(P_1) \wedge$ *initial*$(P_2))$.
- $\approx_{\mathrm{FB:ps}}$ is the coarsest congruence w.r.t. $+$ contained in $\approx_{\mathrm{FB}}$:
  - $P_1 \approx_{\mathrm{FB:ps}} P_2$ iff $P_1 + P \approx_{\mathrm{FB}} P_2 + P$
    for all $P \in \mathbb{P}$ s.t. *initial*$(P) \vee ($*initial*$(P_1) \wedge$ *initial*$(P_2))$.
- $\approx_{\mathrm{FRB:ps}}$ is the coarsest congruence w.r.t. $+$ contained in $\approx_{\mathrm{FRB}}$:
  - $P_1 \approx_{\mathrm{FRB:ps}} P_2$ iff $P_1 + P \approx_{\mathrm{FRB}} P_2 + P$
    for all $P \in \mathbb{P}$ s.t. *initial*$(P) \vee ($*initial*$(P_1) \wedge$ *initial*$(P_2))$.

Modal Logic Characterizations

- Modal logic with weak forward/backward modalities ($a \in \mathcal{A} \setminus \{\tau\}$):

$$\phi \ ::= \ \mathsf{true} \mid \mathsf{init} \mid \neg\phi \mid \phi \wedge \phi \mid \langle\langle\tau\rangle\rangle\phi \mid \langle\langle a\rangle\rangle\phi \mid \langle\langle\tau^{\dagger}\rangle\rangle\phi \mid \langle\langle a^{\dagger}\rangle\rangle\phi$$

Modal Logic Characterizations

- Modal logic with weak forward/backward modalities ($a \in \mathcal{A} \setminus \{\tau\}$):

$$\phi \ ::= \ \mathsf{true} \mid \mathsf{init} \mid \neg\phi \mid \phi \wedge \phi \mid \langle\!\langle \tau \rangle\!\rangle \phi \mid \langle\!\langle a \rangle\!\rangle \phi \mid \langle\!\langle \tau^{\dagger} \rangle\!\rangle \phi \mid \langle\!\langle a^{\dagger} \rangle\!\rangle \phi$$

- Semantics:

$$\begin{array}{lcll}
P & \models & \mathsf{true} & \text{for all } P \in \mathbb{P} \\
P & \models & \mathsf{init} & \text{iff } \mathit{initial}(P) \\
P & \models & \neg\phi & \text{iff } P \not\models \phi \\
P & \models & \phi_1 \wedge \phi_2 & \text{iff } P \models \phi_1 \text{ and } P \models \phi_2 \\
P & \models & \langle\!\langle \tau \rangle\!\rangle \phi & \text{iff there exists } P \Longrightarrow P' \text{ such that } P' \models \phi \\
P & \models & \langle\!\langle a \rangle\!\rangle \phi & \text{iff there exists } P \Longrightarrow \xrightarrow{\theta} \Longrightarrow P' \text{ s.t. } \mathit{act}(\theta) = a \text{ and } P' \models \phi \\
P & \models & \langle\!\langle \tau^{\dagger} \rangle\!\rangle \phi & \text{iff there exists } P' \Longrightarrow P \text{ such that } P' \models \phi \\
P & \models & \langle\!\langle a^{\dagger} \rangle\!\rangle \phi & \text{iff there exists } P' \Longrightarrow \xrightarrow{\theta} \Longrightarrow P \text{ s.t. } \mathit{act}(\theta) = a \text{ and } P' \models \phi
\end{array}$$

- Fragments characterizing the five weak bisimilarities:

	true	init	$\neg$	$\wedge$	$\langle\langle \tau \rangle\rangle$	$\langle\langle a \rangle\rangle$	$\langle\langle \tau^{\dagger} \rangle\rangle$	$\langle\langle a^{\dagger} \rangle\rangle$
$\mathcal{L}^{\tau}_{\mathrm{FB}}$	✓		✓	✓	✓	✓		
$\mathcal{L}^{\tau}_{\mathrm{FB:ps}}$	✓	✓	✓	✓	✓	✓		
$\mathcal{L}^{\tau}_{\mathrm{RB}}$	✓						✓	✓
$\mathcal{L}^{\tau}_{\mathrm{FRB}}$	✓		✓	✓	✓	✓	✓	✓
$\mathcal{L}^{\tau}_{\mathrm{FRB:ps}}$	✓	✓	✓	✓	✓	✓	✓	✓

- $\mathcal{L}^{\tau}_{\mathrm{FB}}$ / $\mathcal{L}^{\tau}_{\mathrm{FB:ps}}$ / $\mathcal{L}^{\tau}_{\mathrm{RB}}$ / $\mathcal{L}^{\tau}_{\mathrm{FRB}}$ / $\mathcal{L}^{\tau}_{\mathrm{FRB:ps}}$ characterizes $\approx_{\mathrm{FB}}$ / $\approx_{\mathrm{FB:ps}}$ / $\approx_{\mathrm{RB}}$ / $\approx_{\mathrm{FRB}}$ / $\approx_{\mathrm{FRB:ps}}$:

$$P_1 \approx_B P_2 \text{ iff } \forall \phi \in \mathcal{L}^{\tau}_{B}.\, P_1 \models \phi \Longleftrightarrow P_2 \models \phi$$

Equational Characterizations

- Additional operator-specific axioms called τ-laws:

(A_1^τ)	$[\approx_{\mathrm{FB:ps}}]$	$a \,.\, \tau \,.\, P$	$=$	$a \,.\, P$	where $\mathit{initial}(P)$
(A_2^τ)	$[\approx_{\mathrm{FB:ps}}]$	$P + \tau \,.\, P$	$=$	$\tau \,.\, P$	where $\mathit{initial}(P)$
(A_3^τ)	$[\approx_{\mathrm{FB:ps}}]$	$a \,.\, (P + \tau \,.\, Q) + a \,.\, Q$	$=$	$a \,.\, (P + \tau \,.\, Q)$	where P, Q initial
(A_4^τ)	$[\approx_{\mathrm{FB:ps}}]$	$a^\dagger \,.\, \tau \,.\, P$	$=$	$a^\dagger \,.\, P$	where $\mathit{initial}(P)$
(A_5^τ)	$[\approx_{\mathrm{RB}}]$	$\tau^\dagger \,.\, P$	$=$	P	
(A_6^τ)	$[\approx_{\mathrm{FRB:ps}}]$	$a \,.\, (\tau \,.\, (P + Q) + P)$	$=$	$a \,.\, (P + Q)$	where P, Q initial
(A_7^τ)	$[\approx_{\mathrm{FRB:ps}}]$	$a^\dagger \,.\, (\tau \,.\, (P + Q) + P')$	$=$	$a^\dagger \,.\, (P' + Q)$	if $\mathit{to\_initial}(P') = P$, where P, Q initial
(A_8^τ)	$[\approx_{\mathrm{FRB:ps}}]$	$a^\dagger \,.\, (\tau^\dagger \,.\, (P' + Q) + P)$	$=$	$a^\dagger \,.\, (P' + Q)$	if $\mathit{to\_initial}(P') = P$, where $\mathit{initial}(P)$

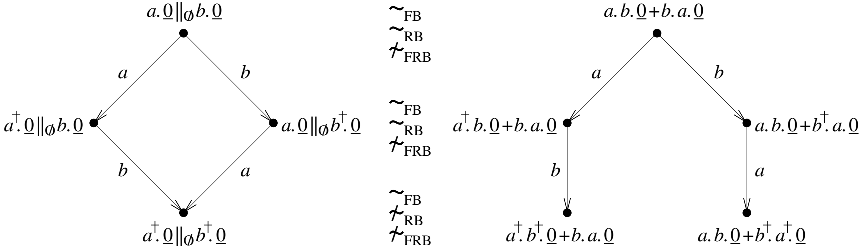
- A_1^τ, A_2^τ, A_3^τ are Milner τ-laws, A_4^τ needed for completeness.
- A_5^τ is a variant of $\tau \,.\, P = P$ (not valid for weak bisim. congruence).
- A_6^τ is Van Glabbeek-Weijland τ-law, A_7^τ and A_8^τ needed for complet.
- $\vdash^{1,2,3,4,5,6,9}_{1,2,3,4}$ / $\vdash^{1,2,7,8}_{5}$ / $\vdash^{1,2,3,10}_{6,7,8}$ is sound and complete for $\approx_{\mathrm{FB:ps}}$ / $\approx_{\mathrm{RB}}$ / $\approx_{\mathrm{FRB:ps}}$ over renaming-free sequential processes.
- $\approx_{\mathrm{FRB}}$ is branching bisimilarity over initial sequential processes!

Expansion Laws for Parallel Composition

- In forward-only process calculi $a \,.\, \underline{0} \,\|_{\emptyset}\, b \,.\, \underline{0}$ and $a \,.\, b \,.\, \underline{0} + b \,.\, a \,.\, \underline{0}$ are deemed equivalent: *the latter is the expansion of the former*.

Expansion Laws for Parallel Composition

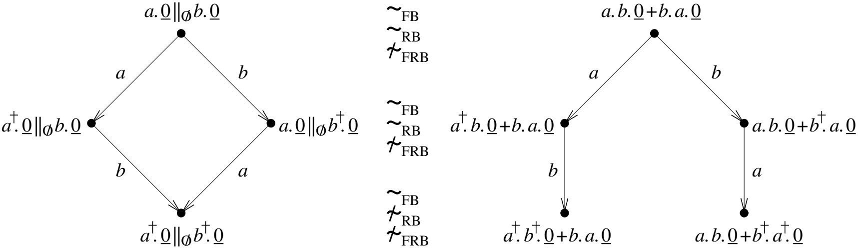
- In forward-only process calculi $a \,.\, \underline{0} \parallel_{\emptyset} b \,.\, \underline{0}$ and $a \,.\, b \,.\, \underline{0} + b \,.\, a \,.\, \underline{0}$ are deemed equivalent: *the latter is the expansion of the former.*
- In our reversible setting we obtain instead $(a \neq b)$:



- $\sim_{\mathrm{FB}}$ is interleaving, while $\sim_{\mathrm{RB}}$ and $\sim_{\mathrm{FRB}}$ are truly concurrent.

Expansion Laws for Parallel Composition

- In forward-only process calculi $a \,.\, \underline{0} \parallel_{\emptyset} b \,.\, \underline{0}$ and $a \,.\, b \,.\, \underline{0} + b \,.\, a \,.\, \underline{0}$ are deemed equivalent: *the latter is the expansion of the former*.
- In our reversible setting we obtain instead $(a \neq b)$:



- $\sim_{\mathrm{FB}}$ is interleaving, while $\sim_{\mathrm{RB}}$ and $\sim_{\mathrm{FRB}}$ are truly concurrent.
- What are the expansion laws for the six bisimulation congruences $\sim_{\mathrm{FB:ps}}$, $\sim_{\mathrm{RB}}$, $\sim_{\mathrm{FRB}}$, $\approx_{\mathrm{FB:ps}}$, $\approx_{\mathrm{RB}}$, $\approx_{\mathrm{FRB:ps}}$?

- Expansion laws for forward-only calculi in the interleaving setting are used to identify $a\,.\,\underline{0} \parallel_{\emptyset} b\,.\,\underline{0}$ and $a\,.\,b\,.\,\underline{0} + b\,.\,a\,.\,\underline{0}$.

- Expansion laws for forward-only calculi in the interleaving setting are used to identify $a \,.\, \underline{0} \parallel_{\emptyset} b \,.\, \underline{0}$ and $a \,.\, b \,.\, \underline{0} + b \,.\, a \,.\, \underline{0}$.
- Also used in truly concurrent semantics to distinguish those processes by adding suitable discriminating information within action prefixes:

- Expansion laws for forward-only calculi in the interleaving setting are used to identify $a \,.\, \underline{0} \,\|_\emptyset\, b \,.\, \underline{0}$ and $a \,.\, b \,.\, \underline{0} + b \,.\, a \,.\, \underline{0}$.
- Also used in truly concurrent semantics to distinguish those processes by adding suitable discriminating information within action prefixes:
  - Causal bisimilarity [DarondeauDegano90] (corresponding to history-preserving bisimilarity [RabinovichTrakhtenbrot88]): every action is enriched with the set of its causing actions each of which is expressed as a numeric backward pointer, hence we get $<a, \emptyset> \,.\, <b, \emptyset> \,.\, \underline{0} + <b, \emptyset> \,.\, <a, \emptyset> \,.\, \underline{0}$ and $<a, \emptyset> \,.\, <b, \{1\}> \,.\, \underline{0} + <b, \emptyset> \,.\, <a, \{1\}> \,.\, \underline{0}$.

- Expansion laws for forward-only calculi in the interleaving setting are used to identify $a \,.\, \underline{0} \,\|_{\emptyset}\, b \,.\, \underline{0}$ and $a \,.\, b \,.\, \underline{0} + b \,.\, a \,.\, \underline{0}$.
- Also used in truly concurrent semantics to distinguish those processes by adding suitable discriminating information within action prefixes:
  - Causal bisimilarity [DarondeauDegano90] (corresponding to history-preserving bisimilarity [RabinovichTrakhtenbrot88]): every action is enriched with the set of its causing actions each of which is expressed as a numeric backward pointer, hence we get $<a, \emptyset> . <b, \emptyset> . \underline{0} + <b, \emptyset> . <a, \emptyset> . \underline{0}$ and $<a, \emptyset> . <b, \{1\}> . \underline{0} + <b, \emptyset> . <a, \{1\}> . \underline{0}$.
  - Location bisimilarity [BoudolCastellaniHennessyKiehn94]: every action is enriched with the name of the location in which it is executed, hence we get $<a, l_a> . <b, l_b> . \underline{0} + <b, l_b> . <a, l_a> . \underline{0}$ and $<a, l_a> . <b, l_a l_b> . \underline{0} + <b, l_b> . <a, l_b l_a> . \underline{0}$.

- Expansion laws for forward-only calculi in the interleaving setting are used to identify $a\,.\,\underline{0} \parallel_{\emptyset} b\,.\,\underline{0}$ and $a\,.\,b\,.\,\underline{0} + b\,.\,a\,.\,\underline{0}$.
- Also used in truly concurrent semantics to distinguish those processes by adding suitable discriminating information within action prefixes:
  - Causal bisimilarity [DarondeauDegano90] (corresponding to history-preserving bisimilarity [RabinovichTrakhtenbrot88]): every action is enriched with the set of its causing actions each of which is expressed as a numeric backward pointer, hence we get $<a, \emptyset>\,.\,<b, \emptyset>\,.\,\underline{0} + <b, \emptyset>\,.\,<a, \emptyset>\,.\,\underline{0}$ and $<a, \emptyset>\,.\,<b, \{1\}>\,.\,\underline{0} + <b, \emptyset>\,.\,<a, \{1\}>\,.\,\underline{0}$.
  - Location bisimilarity [BoudolCastellaniHennessyKiehn94]: every action is enriched with the name of the location in which it is executed, hence we get $<a, l_a>\,.\,<b, l_b>\,.\,\underline{0} + <b, l_b>\,.\,<a, l_a>\,.\,\underline{0}$ and $<a, l_a>\,.\,<b, l_a l_b>\,.\,\underline{0} + <b, l_b>\,.\,<a, l_b l_a>\,.\,\underline{0}$.
  - Pomset bisimilarity [BoudolCastellani88]: a prefix may contain a combination of actions that are causally related or independent, hence the former process becomes $a\,.\,b\,.\,\underline{0} + b\,.\,a\,.\,\underline{0} + (a \parallel b)\,.\,\underline{0}$.

- How to uniformly derive the six expansion laws?
- Proved trees approach of [DeganoPriami92].
- Enabled by the use of proof terms in the operational semantics!

- How to uniformly derive the six expansion laws?
- Proved trees approach of [DeganoPriami92].
- Enabled by the use of proof terms in the operational semantics!
- Interleaving: proof terms are reduced to the actions they contain, hence we are done for $\sim_{\mathrm{FB:ps}}$ and $\approx_{\mathrm{FB:ps}}$.

- How to uniformly derive the six expansion laws?
- Proved trees approach of [DeganoPriami92].
- Enabled by the use of proof terms in the operational semantics!
- Interleaving: proof terms are reduced to the actions they contain, hence we are done for $\sim_{\mathrm{FB:ps}}$ and $\approx_{\mathrm{FB:ps}}$.
- True concurrency: they are transformed into actions extended with suitable discriminating information (then encode processes accordingly).
- Information already available in the operational semantics for causal bisimilarity, location bisimilarity, pomset bisimilarity.
- Unfortunately not available in our proved operational semantics for $\sim_{\mathrm{RB}}$, $\sim_{\mathrm{FRB}}$, $\approx_{\mathrm{RB}}$, $\approx_{\mathrm{FRB:ps}}$!

- The equivalence of interest drives an observation function that maps proof terms to the required observations.
- Observation function ℓ applied to proof terms labeling transitions, so that $\ell(\theta_1)$ and $\ell(\theta_2)$ are considered in the bisimulation game.
- Action preservation: $\ell(\theta_1) = \ell(\theta_2)$ implies $\mathit{act}(\theta_1) = \mathit{act}(\theta_2)$.
- ℓ may depend on other possible parameters that are present in the proved labeled transition system.
- $\sim_{\mathrm{FB:ps:}\ell_{\mathrm{F}}}$, $\sim_{\mathrm{RB:}\ell_{\mathrm{R}}}$, $\sim_{\mathrm{FRB:}\ell_{\mathrm{FR}}}$, $\approx_{\mathrm{FB:ps:}\ell_{\mathrm{F,w}}}$, $\approx_{\mathrm{RB:}\ell_{\mathrm{R,w}}}$, $\approx_{\mathrm{FRB:ps:}\ell_{\mathrm{FR,w}}}$ are the six resulting equivalences.
- When do they coincide with the six congruences?
- What is the discriminating information needed by reverse and forward-reverse semantics?

- As already anticipated $\sim_{\rm FB:ps:\ell_F} = \sim_{\rm FB:ps}$ and $\approx_{\rm FB:ps:\ell_{F,w}} = \approx_{\rm FB:ps}$ when $\ell_{\rm F}(\theta) = \ell_{\rm F,w}(\theta) = \mathit{act}(\theta)$.
- Expansion law for $\sim_{\rm FB:ps}$ and $\approx_{\rm FB:ps}$:

$$(\mathsf{A}_{15}) \quad P_1 \parallel_L P_2 \;=\; [a^{\dagger}.] \Big(\sum_{i \in I_1, a_{1,i} \notin L} a_{1,i} \,.\, (P_{1,i} \parallel_L P'_2) + \sum_{i \in I_2, a_{2,i} \notin L} a_{2,i} \,.\, (P'_1 \parallel_L P_{2,i}) + \sum_{i \in I_1, a_{1,i} \in L} \; \sum_{j \in I_2, a_{2,j} = a_{1,i}} a_{1,i} \,.\, (P_{1,i} \parallel_L P_{2,j}) \Big)$$

- $P_k = [a_k^{\dagger}.]P'_k$ with $P'_k = \sum_{i \in I_k} a_{k,i} \,.\, P_{k,i}$ for $k \in \{1, 2\}$, called F-nf.
- $[a^{\dagger}.]$ is present iff $[a_1^{\dagger}.]$ or $[a_2^{\dagger}.]$ is present (they are optional).

- $\sim_{\mathrm{RB}:\ell_{\mathrm{R}}} = \sim_{\mathrm{RB}}$ and $\sim_{\mathrm{FRB}:\ell_{\mathrm{FR}}} = \sim_{\mathrm{FRB}}$ when $\ell_{\mathrm{R}}(\theta)_{P'} = \ell_{\mathrm{FR}}(\theta)_{P'} = <\mathit{act}(\theta), \mathit{brs}(P')> \triangleq \ell_{\mathrm{brs}}(\theta)_{P'}$ for every proved transition $P \xrightarrow{\theta} P'$.
- $\mathit{brs}(P')$ is the backward ready set of P', the set of actions labeling the incoming transitions of P'.

- $\sim_{\mathrm{RB}:\ell_{\mathrm{R}}} = \sim_{\mathrm{RB}}$ and $\sim_{\mathrm{FRB}:\ell_{\mathrm{FR}}} = \sim_{\mathrm{FRB}}$ when $\ell_{\mathrm{R}}(\theta)_{P'} = \ell_{\mathrm{FR}}(\theta)_{P'} = \ <\mathit{act}(\theta), \mathit{brs}(P')> \triangleq \ell_{\mathrm{brs}}(\theta)_{P'}$ for every proved transition $P \xrightarrow{\theta} P'$.
- $\mathit{brs}(P')$ is the backward ready set of P', the set of actions labeling the incoming transitions of P'.
- Thus $a\,.\,\underline{0} \,\|_{\emptyset}\, b\,.\,\underline{0}$ is encoded as:

  $<a, \{a\}>\,.\,<b, \{a, b\}>\,.\,\underline{0} + <b, \{b\}>\,.\,<a, \{a, b\}>\,.\,\underline{0}$

  while $a\,.\,b\,.\,\underline{0} + b\,.\,a\,.\,\underline{0}$ is encoded as:

  $<a, \{a\}>\,.\,<b, \{b\}>\,.\,\underline{0} + <b, \{b\}>\,.\,<a, \{a\}>\,.\,\underline{0}$

- $\sim_{\mathrm{RB}:\ell_{\mathrm{R}}} = \sim_{\mathrm{RB}}$ and $\sim_{\mathrm{FRB}:\ell_{\mathrm{FR}}} = \sim_{\mathrm{FRB}}$ when $\ell_{\mathrm{R}}(\theta)_{P'} = \ell_{\mathrm{FR}}(\theta)_{P'} = {<}\mathit{act}(\theta), \mathit{brs}(P'){>} \triangleq \ell_{\mathrm{brs}}(\theta)_{P'}$ for every proved transition $P \xrightarrow{\theta} P'$.
- $\mathit{brs}(P')$ is the backward ready set of P', the set of actions labeling the incoming transitions of P'.
- Thus $a \,.\, \underline{0} \,\|_{\emptyset}\, b \,.\, \underline{0}$ is encoded as:

  $${<}a, \{a\}{>} \,.\, {<}b, \{a, b\}{>} \,.\, \underline{0} + {<}b, \{b\}{>} \,.\, {<}a, \{a, b\}{>} \,.\, \underline{0}$$

  while $a \,.\, b \,.\, \underline{0} + b \,.\, a \,.\, \underline{0}$ is encoded as:

  $${<}a, \{a\}{>} \,.\, {<}b, \{b\}{>} \,.\, \underline{0} + {<}b, \{b\}{>} \,.\, {<}a, \{a\}{>} \,.\, \underline{0}$$
- The encoding of $a^{\dagger} \,.\, \underline{0} \,\|_{\emptyset}\, b^{\dagger} \,.\, \underline{0}$ (a case not addressed in [DP92]) *cannot be*:

  $${<}a^{\dagger}, \{a\}{>} \,.\, {<}b^{\dagger}, \{a, b\}{>} \,.\, \underline{0} + {<}b^{\dagger}, \{b\}{>} \,.\, {<}a^{\dagger}, \{a, b\}{>} \,.\, \underline{0}$$
- It is ${<}a^{\dagger}, \{a\}{>} \,.\, {<}b^{\dagger}, \{a, b\}{>} \,.\, \underline{0} + {<}b, \{b\}{>} \,.\, {<}a, \{a, b\}{>} \,.\, \underline{0}$ or ${<}a, \{a\}{>} \,.\, {<}b, \{a, b\}{>} \,.\, \underline{0} + {<}b^{\dagger}, \{b\}{>} \,.\, {<}a^{\dagger}, \{a, b\}{>} \,.\, \underline{0}$ depending on whether trace $a\,b$ or trace $b\,a$ has been executed (initial subprocesses are needed by the forward-reverse semantics).

- Encoding to $\mathbb{P}_{\mathrm{brs}}$: set of sequential processes in which every action prefix is a pair composed of an action and an action set.
- Let $\widetilde{P}$ be the ℓ_{brs}-encoding of P.
- Let $\widehat{P}$ be the $\ell_{\mathrm{brs,w}}$-encoding of P.

- Encoding to $\mathbb{P}_{\mathrm{brs}}$: set of sequential processes in which every action prefix is a pair composed of an action and an action set.
- Let $\widetilde{P}$ be the ℓ_{brs}-encoding of P.
- Let $\widehat{P}$ be the $\ell_{\mathrm{brs,w}}$-encoding of P.
- Expansion laws for $\sim_{\mathrm{RB}}$ and $\approx_{\mathrm{RB}}$:

$$
\begin{array}{lrcl}
(\mathsf{A}_{16}) & \widetilde{P_1 \parallel_L P_2} & = & e\ell^{\varepsilon}_{\mathrm{brs,R}}(\widetilde{P}_1, \widetilde{P}_2, L)_{P_1 \parallel_L P_2} \\
(\mathsf{A}_{17}) & \widehat{P_1 \parallel_L P_2} & = & e\ell^{\varepsilon}_{\mathrm{brs,R}}(\widehat{P}_1, \widehat{P}_2, L)_{P_1 \parallel_L P_2}
\end{array}
$$

- $P_k = \underline{0}$ or $P_k = a^{\dagger} . P'_k$ for $k \in \{1, 2\}$, called R-nf.

- Encoding to $\mathbb{P}_{\rm brs}$: set of sequential processes in which every action prefix is a pair composed of an action and an action set.
- Let $\widetilde{P}$ be the $\ell_{\rm brs}$-encoding of P.
- Let $\widehat{P}$ be the $\ell_{\rm brs,w}$-encoding of P.
- Expansion laws for $\sim_{\rm RB}$ and $\approx_{\rm RB}$:

$$
\begin{array}{lrcl}
(\mathrm{A}_{16}) & \widetilde{P_1 \parallel_L P_2} & = & e\ell^{\varepsilon}_{\rm brs,R}(\widetilde{P}_1, \widetilde{P}_2, L)_{P_1 \parallel_L P_2} \\
(\mathrm{A}_{17}) & \widehat{P_1 \parallel_L P_2} & = & e\ell^{\varepsilon}_{\rm brs,R}(\widehat{P}_1, \widehat{P}_2, L)_{P_1 \parallel_L P_2}
\end{array}
$$

- $P_k = \underline{0}$ or $P_k = a^{\dagger} \,.\, P'_k$ for $k \in \{1, 2\}$, called R-nf.
- Expansion laws for $\sim_{\rm FRB}$ and $\approx_{\rm FRB:ps}$:

$$
\begin{array}{lrcl}
(\mathrm{A}_{18}) & \widetilde{P_1 \parallel_L P_2} & = & e\ell^{\varepsilon}_{\rm brs}(\widetilde{P}_1, \widetilde{P}_2, L)_{P_1 \parallel_L P_2} \\
(\mathrm{A}_{19}) & \widehat{P_1 \parallel_L P_2} & = & e\ell^{\varepsilon}_{\rm brs}(\widehat{P}_1, \widehat{P}_2, L)_{P_1 \parallel_L P_2}
\end{array}
$$

- $P_k = [a^{\dagger} \,.\, P'_k +] \sum_{i \in I_k} a_{k,i} \,.\, P_{k,i}$ for $k \in \{1, 2\}$, called FR-nf.

True Concurrency

- How close is $\sim_{\mathrm{FRB}}$ to hereditary history-preserving bisimilarity?

True Concurrency

- How close is $\sim_{\mathrm{FRB}}$ to hereditary history-preserving bisimilarity?
- A labeled configuration structure is a tuple $C = (\mathcal{E}, \mathcal{C}, \ell)$ where:
  - $\mathcal{E}$ is a set of events.
  - $\mathcal{C} \subseteq \mathcal{P}_{\mathrm{fin}}(\mathcal{E})$ is a set of configurations.
  - $\ell : \bigcup_{X \in \mathcal{C}} X \to \mathcal{A}$ is the labeling function.

True Concurrency

- How close is $\sim_{\mathrm{FRB}}$ to hereditary history-preserving bisimilarity?
- A labeled configuration structure is a tuple $C = (\mathcal{E}, \mathcal{C}, \ell)$ where:
  - $\mathcal{E}$ is a set of events.
  - $\mathcal{C} \subseteq \mathcal{P}_{\mathrm{fin}}(\mathcal{E})$ is a set of configurations.
  - $\ell : \bigcup_{X \in \mathcal{C}} X \to \mathcal{A}$ is the labeling function.
- A configuration structure C is stable iff it is:
  - Rooted: $\emptyset \in \mathcal{C}$.
  - Connected: $\forall X \in \mathcal{C} \setminus \{\emptyset\}. \exists e \in X. X \setminus \{e\} \in \mathcal{C}$.
  - Closed under bounded unions and intersections:
    $\forall X, Y, Z \in \mathcal{C}. X \cup Y \subseteq Z \implies X \cup Y \in \mathcal{C} \wedge X \cap Y \in \mathcal{C}$.

True Concurrency

- How close is $\sim_{\mathrm{FRB}}$ to hereditary history-preserving bisimilarity?
- A labeled configuration structure is a tuple $C = (\mathcal{E}, \mathcal{C}, \ell)$ where:
  - $\mathcal{E}$ is a set of events.
  - $\mathcal{C} \subseteq \mathcal{P}_{\mathrm{fin}}(\mathcal{E})$ is a set of configurations.
  - $\ell : \bigcup_{X \in \mathcal{C}} X \to \mathcal{A}$ is the labeling function.
- A configuration structure C is stable iff it is:
  - Rooted: $\emptyset \in \mathcal{C}$.
  - Connected: $\forall X \in \mathcal{C} \setminus \{\emptyset\}. \exists e \in X. X \setminus \{e\} \in \mathcal{C}$.
  - Closed under bounded unions and intersections:
    $\forall X, Y, Z \in \mathcal{C}. X \cup Y \subseteq Z \implies X \cup Y \in \mathcal{C} \wedge X \cap Y \in \mathcal{C}$.
- The causality relation over $X \in \mathcal{C}$ is defined by letting $e_1 <_X e_2$ for $e_1, e_2 \in X$ s.t. $e_1 \neq e_2$ iff $\forall Y \in \mathcal{C}. Y \subseteq X \wedge e_2 \in Y \implies e_1 \in Y$.
- The concurrency relation over X is $co_X = (X \times X) \setminus (\leq_X \cup \geq_X)$.

True Concurrency

- How close is $\sim_{\mathrm{FRB}}$ to hereditary history-preserving bisimilarity?
- A labeled configuration structure is a tuple $C = (\mathcal{E}, \mathcal{C}, \ell)$ where:
  - $\mathcal{E}$ is a set of events.
  - $\mathcal{C} \subseteq \mathcal{P}_{\mathrm{fin}}(\mathcal{E})$ is a set of configurations.
  - $\ell : \bigcup_{X \in \mathcal{C}} X \to \mathcal{A}$ is the labeling function.
- A configuration structure C is stable iff it is:
  - Rooted: $\emptyset \in \mathcal{C}$.
  - Connected: $\forall X \in \mathcal{C} \setminus \{\emptyset\}. \exists e \in X. X \setminus \{e\} \in \mathcal{C}$.
  - Closed under bounded unions and intersections: $\forall X, Y, Z \in \mathcal{C}. X \cup Y \subseteq Z \implies X \cup Y \in \mathcal{C} \wedge X \cap Y \in \mathcal{C}$.
- The causality relation over $X \in \mathcal{C}$ is defined by letting $e_1 <_X e_2$ for $e_1, e_2 \in X$ s.t. $e_1 \neq e_2$ iff $\forall Y \in \mathcal{C}. Y \subseteq X \wedge e_2 \in Y \implies e_1 \in Y$.
- The concurrency relation over X is $co_X = (X \times X) \setminus (\leq_X \cup \geq_X)$.
- $X \xrightarrow{a} X'$ for $X, X' \in \mathcal{C}$ iff $X \subseteq X' \wedge X' \setminus X = \{e\} \wedge \ell(e) = a$.

- Two stable configuration structures $C_i = (\mathcal{E}_i, \mathcal{C}_i, l_i)$, $i \in \{1, 2\}$, are hereditary history-preserving bisimilar, written $C_1 \sim_{\mathrm{HHPB}} C_2$, iff there exists a hereditary history-preserving bisimulation between C_1 and C_2, i.e., a relation $\mathcal{B} \subseteq \mathcal{C}_1 \times \mathcal{C}_2 \times \mathcal{P}(\mathcal{E}_1 \times \mathcal{E}_2)$ such that:

- Two stable configuration structures $C_i = (\mathcal{E}_i, \mathcal{C}_i, l_i)$, $i \in \{1, 2\}$, are hereditary history-preserving bisimilar, written $C_1 \sim_{\mathrm{HHPB}} C_2$, iff there exists a hereditary history-preserving bisimulation between C_1 and C_2, i.e., a relation $\mathcal{B} \subseteq \mathcal{C}_1 \times \mathcal{C}_2 \times \mathcal{P}(\mathcal{E}_1 \times \mathcal{E}_2)$ such that:
  - $(\emptyset, \emptyset, \emptyset) \in \mathcal{B}$.

- Two stable configuration structures $C_i = (\mathcal{E}_i, \mathcal{C}_i, l_i)$, $i \in \{1, 2\}$, are hereditary history-preserving bisimilar, written $C_1 \sim_{\mathrm{HHPB}} C_2$, iff there exists a hereditary history-preserving bisimulation between C_1 and C_2, i.e., a relation $\mathcal{B} \subseteq \mathcal{C}_1 \times \mathcal{C}_2 \times \mathcal{P}(\mathcal{E}_1 \times \mathcal{E}_2)$ such that:
  - $(\emptyset, \emptyset, \emptyset) \in \mathcal{B}$.
  - Whenever $(X_1, X_2, f) \in \mathcal{B}$, then:
    - f is a *bijection* from X_1 to X_2 that preserves *labeling*, i.e., $l_1(e) = l_2(f(e))$ for all $e \in X_1$, and *causality*, i.e., $e \leq_{X_1} e' \iff f(e) \leq_{X_2} f(e')$ for all $e, e' \in X_1$.

- Two stable configuration structures $C_i = (\mathcal{E}_i, \mathcal{C}_i, l_i)$, $i \in \{1, 2\}$, are hereditary history-preserving bisimilar, written $C_1 \sim_{\mathrm{HHPB}} C_2$, iff there exists a hereditary history-preserving bisimulation between C_1 and C_2, i.e., a relation $\mathcal{B} \subseteq \mathcal{C}_1 \times \mathcal{C}_2 \times \mathcal{P}(\mathcal{E}_1 \times \mathcal{E}_2)$ such that:
  - $(\emptyset, \emptyset, \emptyset) \in \mathcal{B}$.
  - Whenever $(X_1, X_2, f) \in \mathcal{B}$, then:
    - f is a *bijection* from X_1 to X_2 that preserves *labeling*, i.e., $l_1(e) = l_2(f(e))$ for all $e \in X_1$, and *causality*, i.e., $e \leq_{X_1} e' \iff f(e) \leq_{X_2} f(e')$ for all $e, e' \in X_1$.
    - For each $X_1 \xrightarrow{a}_{C_1} X_1'$ there exist $X_2 \xrightarrow{a}_{C_2} X_2'$ and f' such that $(X_1', X_2', f') \in \mathcal{B}$ and $f' \upharpoonright X_1 = f$, and vice versa.

- Two stable configuration structures $C_i = (\mathcal{E}_i, \mathcal{C}_i, l_i)$, $i \in \{1, 2\}$, are hereditary history-preserving bisimilar, written $C_1 \sim_{\mathrm{HHPB}} C_2$, iff there exists a hereditary history-preserving bisimulation between C_1 and C_2, i.e., a relation $\mathcal{B} \subseteq \mathcal{C}_1 \times \mathcal{C}_2 \times \mathcal{P}(\mathcal{E}_1 \times \mathcal{E}_2)$ such that:
  - $(\emptyset, \emptyset, \emptyset) \in \mathcal{B}$.
  - Whenever $(X_1, X_2, f) \in \mathcal{B}$, then:
    - f is a *bijection* from X_1 to X_2 that preserves *labeling*, i.e., $l_1(e) = l_2(f(e))$ for all $e \in X_1$, and *causality*, i.e., $e \leq_{X_1} e' \iff f(e) \leq_{X_2} f(e')$ for all $e, e' \in X_1$.
    - For each $X_1 \xrightarrow{a}_{C_1} X_1'$ there exist $X_2 \xrightarrow{a}_{C_2} X_2'$ and f' such that $(X_1', X_2', f') \in \mathcal{B}$ and $f' \upharpoonright X_1 = f$, and vice versa.
    - For each $X_1' \xrightarrow{a}_{C_1} X_1$ there exist $X_2' \xrightarrow{a}_{C_2} X_2$ and f' such that $(X_1', X_2', f') \in \mathcal{B}$ and $f \upharpoonright X_1' = f'$, and vice versa.

- $\sim_{\mathrm{HHPB}}$ [Bednarczyk91] is the finest truly concurrent equivalence preserved under action refinement that is capable of respecting causality, branching, and their interplay while abstracting from choices between identical alternatives [VanGlabbeekGoltz01].
- $\sim_{\mathrm{FRB}}$ coincides with $\sim_{\mathrm{HHPB}}$ in the absence of autoconcurrency at the same causality level [PhillipsUlidowski12].
- Cross fertilization for their equational and logical characterizations.

- $\sim_{\mathrm{HHPB}}$ [Bednarczyk91] is the finest truly concurrent equivalence preserved under action refinement that is capable of respecting causality, branching, and their interplay while abstracting from choices between identical alternatives [VanGlabbeekGoltz01].
- $\sim_{\mathrm{FRB}}$ coincides with $\sim_{\mathrm{HHPB}}$ in the absence of autoconcurrency at the same causality level [PhillipsUlidowski12].
- Cross fertilization for their equational and logical characterizations.
- Autoconcurrency is $a \,.\, \underline{0} \,\|_{\emptyset}\, a \,.\, \underline{0}$, while $a \,.\, a \,.\, \underline{0}$ is autocausation.
- $a \,.\, \underline{0} \,\|_{\emptyset}\, a \,.\, \underline{0} \;\sim_{\mathrm{FRB}}\; a \,.\, a \,.\, \underline{0} + a \,.\, a \,.\, \underline{0} \;\sim_{\mathrm{FRB}}\; a \,.\, a \,.\, \underline{0}$.
- Their ℓ_{brs}-encodings are basically the same:

$$<a, \{a\}> \,.\, <a, \{a, a\}> \,.\, \underline{0} + <a, \{a\}> \,.\, <a, \{a, a\}> \,.\, \underline{0}$$

$$<a, \{a\}> \,.\, <a, \{a\}> \,.\, \underline{0} + <a, \{a\}> \,.\, <a, \{a\}> \,.\, \underline{0}$$

$$<a, \{a\}> \,.\, <a, \{a\}> \,.\, \underline{0}$$

- Denotational semantics $[\![\_]\!]$ for $\mathbb{P}$ based on configuration structures in which events are proof terms.
- $[\![a\,.\,\underline{0} \parallel_{\emptyset} a\,.\,\underline{0}]\!] \not\sim_{\mathrm{HHPB}} [\![a\,.\,a\,.\,\underline{0}]\!]$ as $\underline{\|}_{\emptyset} a$ and $\underline{\|}_{\emptyset} a$ are independent while a and $._{a} a$ are causally related, hence no bijection exists between the former and the latter that preserves causality.

- Denotational semantics $[\![\_]\!]$ for $\mathbb{P}$ based on configuration structures in which events are proof terms.
- $[\![a \,.\, \underline{0} \parallel_{\emptyset} a \,.\, \underline{0}]\!] \not\sim_{\mathrm{HHPB}} [\![a \,.\, a \,.\, \underline{0}]\!]$ as $\lfloor\!\lfloor_{\emptyset} a$ and $\rfloor\!\rfloor_{\emptyset} a$ are independent while a and $._{a} a$ are causally related, hence no bijection exists between the former and the latter that preserves causality.
- $\sim_{\mathrm{FRB}}$ plus backward ready multiset equality distinguish them.
- $\sim_{\mathrm{FRB:brm}} = \sim_{\mathrm{HHPB}}$ in the presence of autoconcurrency if for each set of conflicting events all those events are caused by the same event.
- $\sim_{\mathrm{FRB:brm}}$ *counts* the incoming a-transitions of related configurations, no bijection between identically labeled events [AubertCristescu20].

- Denotational semantics $[\![\_]\!]$ for $\mathbb{P}$ based on configuration structures in which events are proof terms.
- $[\![a\,.\,\underline{0}\parallel_{\emptyset} a\,.\,\underline{0}]\!] \not\sim_{\mathrm{HHPB}} [\![a\,.\,a\,.\,\underline{0}]\!]$ as $\parallel_{\emptyset} a$ and $\lfloor\!\rfloor_{\emptyset} a$ are independent while a and $._{a} a$ are causally related, hence no bijection exists between the former and the latter that preserves causality.
- $\sim_{\mathrm{FRB}}$ plus backward ready multiset equality distinguish them.
- $\sim_{\mathrm{FRB:brm}} = \sim_{\mathrm{HHPB}}$ in the presence of autoconcurrency if for each set of conflicting events all those events are caused by the same event.
- $\sim_{\mathrm{FRB:brm}}$ *counts* the incoming a-transitions of related configurations, no bijection between identically labeled events [AubertCristescu20].
- $\sim_{\mathrm{FRB:brm}}$ over $\mathbb{P}$ is an operational representation of $\sim_{\mathrm{HHPB}}$.
- The ℓ_{brm}-encoding of $a\,.\,\underline{0}\parallel_{\emptyset} a\,.\,\underline{0}$:

  $$<a, \{\!|\, a \,|\!\}> .\, <a, \{\!|\, a, a \,|\!\}> .\, \underline{0} + <a, \{\!|\, a \,|\!\}> .\, <a, \{\!|\, a, a \,|\!\}> .\, \underline{0}$$

  differs from its ℓ_{brs}-encoding:

  $$<a, \{a\}> .\, <a, \{a, a\}> .\, \underline{0} + <a, \{a\}> .\, <a, \{a, a\}> .\, \underline{0}$$

Concluding Remarks and Future Work

- Reversibility as a bridge between different worlds that retrospectively enlightens concurrency theory:
  - Forward bisimilarity is the usual bisimilarity.
  - Reverse bisimilarity boils down to reverse trace equivalence over $\mathbb{P}_{\rm seq}$.
  - Weak forward-reverse bisimilarity is branching bisimilarity over $\mathbb{P}_{\rm seq}$.
  - Connection with hereditary history-preserving bisimilarity over $\mathbb{P}$.
  - Expansion laws addressing interleaving semantics or true concurrency.

Concluding Remarks and Future Work

- Reversibility as a bridge between different worlds that retrospectively enlightens concurrency theory:
  - Forward bisimilarity is the usual bisimilarity.
  - Reverse bisimilarity boils down to reverse trace equivalence over $\mathbb{P}_{\rm seq}$.
  - Weak forward-reverse bisimilarity is branching bisimilarity over $\mathbb{P}_{\rm seq}$.
  - Connection with hereditary history-preserving bisimilarity over $\mathbb{P}$.
  - Expansion laws addressing interleaving semantics or true concurrency.
- Noninterference analysis of reversible systems (branching bisimilarity) and extensions of causal reversibility by construction [PU07]:
  - Probabilistic processes (alternation with nondeterminism).
  - Deterministically timed processes (time additivity/determinism).
  - Stochastically timed processes (ordinary/exact/strict lumpability, causal reversibility implies time reversibility).

Concluding Remarks and Future Work

- Reversibility as a bridge between different worlds that retrospectively enlightens concurrency theory:
  - Forward bisimilarity is the usual bisimilarity.
  - Reverse bisimilarity boils down to reverse trace equivalence over $\mathbb{P}_{\rm seq}$.
  - Weak forward-reverse bisimilarity is branching bisimilarity over $\mathbb{P}_{\rm seq}$.
  - Connection with hereditary history-preserving bisimilarity over $\mathbb{P}$.
  - Expansion laws addressing interleaving semantics or true concurrency.
- Noninterference analysis of reversible systems (branching bisimilarity) and extensions of causal reversibility by construction [PU07]:
  - Probabilistic processes (alternation with nondeterminism).
  - Deterministically timed processes (time additivity/determinism).
  - Stochastically timed processes (ordinary/exact/strict lumpability, causal reversibility implies time reversibility).
- When does time reversibility imply causal reversibility?
- What changes when admitting irreversible actions or recursion?
- Underpinning reversible concurrent programming languages?
- Unitary transformations in quantum computing are reversible!

Inspiring References

[1] R. Landauer,
"Irreversibility and Heat Generation in the Computing Process",
IBM Journal of Research and Development 5:183–191, 1961.

[2] C.H. Bennett,
"Logical Reversibility of Computation",
IBM Journal of Research and Development 17:525–532, 1973.

[3] R. De Nicola, U. Montanari, F. Vaandrager,
"Back and Forth Bisimulations",
Proc. of CONCUR 1990.

[4] G. Boudol, I. Castellani,
"Flow Models of Distributed Computations: Three Equivalent Semantics for CCS",
Information and Computation 114:247–314, 1994.

[5] V. Danos, J. Krivine,
"Reversible Communicating Systems",
Proc. of CONCUR 2004.

[6] I. Phillips, I. Ulidowski,
"Reversing Algebraic Process Calculi",
Journal of Logic and Algebraic Programming 73:70–96, 2007.

[7] I. Lanese, I. Phillips, I. Ulidowski,
"An Axiomatic Theory for Reversible Computation",
ACM Trans. on Computational Logic 25(2):11:1–11:40, 2024.

[8] F.P. Kelly,
"Reversibility and Stochastic Networks",
John Wiley & Sons, 1979.

[9] A. Marin, S. Rossi,
"On the Relations between Markov Chain Lumpability and Reversibility",
Acta Informatica 54:447–485, 2017.

[10] P. Degano, C. Priami,
"Proved Trees",
Proc. of ICALP 1992.

[11] G. Boudol, I. Castellani,
"A Non-Interleaving Semantics for CCS Based on Proved Transitions",
Fundamenta Informaticae 11:433–452, 1988.

[12] R.J. van Glabbeek, U. Goltz,
"Refinement of Actions and Equivalence Notions for Concurrent Systems",
Acta Informatica 37:229–327, 2001.

[13] Ph. Darondeau, P. Degano,
"Causal Trees: Interleaving + Causality",
Proc. of the LITP Spring School on Theoretical Computer Science, 1990.

[14] G. Boudol, I. Castellani, M. Hennessy, A. Kiehn,
"A Theory of Processes with Localities",
Formal Aspects of Computing 6:165–200, 1994.

[15] G. Boudol, I. Castellani,
"Concurrency and Atomicity",
Theoretical Computer Science 59:25–84, 1988.

[16] A.M. Rabinovich, B.A. Trakhtenbrot,
"Behavior Structures and Nets",
Acta Informatica 11:357–404, 1988.

[17] M.A. Bednarczyk,
"Hereditary History Preserving Bisimulations or What Is the Power of the Future Perfect in Program Logics",
Technical Report, Polish Academy of Sciences, Gdansk, 1991.

[18] I. Phillips, I. Ulidowski,
"A Hierarchy of Reverse Bisimulations on Stable Configuration Structures",
Mathematical Structures in Computer Science 22:333–372, 2012.

[19] C. Aubert, I. Cristescu,
"How Reversibility Can Solve Traditional Questions: The Example of Hereditary History-Preserving Bisimulation",
Proc. of CONCUR 2020.

Our Contributions

[1] M. Bernardo, S. Rossi,
"Reverse Bisimilarity vs. Forward Bisimilarity",
Proc. of FOSSACS 2023.

[2] M. Bernardo, A. Esposito,
"On the Weak Continuation of Reverse Bisimilarity vs. Forward Bisimilarity",
Proc. of ICTCS 2023.

[3] M. Bernardo, A. Esposito,
"Modal Logic Characterizations of Forward, Reverse, and Forward-Reverse Bisimilarities",
Proc. of GANDALF 2023.

[4] M. Bernardo, A. Esposito, C.A. Mezzina,
"Expansion Laws for Forward-Reverse, Forward, and Reverse Bisimilarities via Proved Encodings",
Proc. of EXPRESS/SOS 2024.

[5] M. Bernardo, A. Esposito, C.A. Mezzina,
"Alternative Characterizations of Hereditary History-Preserving Bisimilarity via Backward Ready Multisets",
Proc. of FOSSACS 2025.

[6] A. Esposito, A. Aldini, M. Bernardo,
"Branching Bisimulation Semantics Enables Noninterference Analysis of Reversible Systems",
Proc. of FORTE 2023.

[7] A. Esposito, A. Aldini, M. Bernardo,
"Noninterference Analysis of Reversible Probabilistic Systems",
Proc. of FORTE 2024.

[8] A. Esposito, A. Aldini, M. Bernardo,
"Noninterference Analysis of Stochastically Timed Reversible Systems",
Proc. of FORTE 2025.

[9] A. Esposito, A. Aldini, M. Bernardo,
"Noninterference Analysis of Deterministically Timed Reversible Systems",
Proc. of QEST+FORMATS 2025.

[10] A. Esposito,
"A Process Algebraic Theory of Reversible Concurrent Systems with Applications to Noninterference Analysis",
PhD Thesis, University of Urbino, 2025.

[11] M. Bernardo, C.A. Mezzina,
"Towards Bridging Time and Causal Reversibility",
Proc. of FORTE 2020.

[12] M. Bernardo, C.A. Mezzina,
"Bridging Causal Reversibility and Time Reversibility: A Stochastic Process Algebraic Approach",
Logical Methods in Computer Science 19(2):6:1–6:27, 2023.

[13] M. Bernardo, C.A. Mezzina,
"Causal Reversibility for Timed Process Calculi with Lazy/Eager Durationless Actions and Time Additivity",
Proc. of FORMATS 2023.

[14] M. Bernardo, C.A. Mezzina,
"Reversibility in Process Calculi with Nondeterminism and Probabilities",
Proc. of ICTAC 2024.

[15] M. Bernardo, I. Lanese, A. Marin, C.A. Mezzina, S. Rossi, C. Sacerdoti Coen,
"Causal Reversibility Implies Time Reversibility",
Proc. of QEST 2023.