ALMA MATER STUDIORUM - UNIVERSITÀ DI BOLOGNA

SCUOLA DI LETTERE E BENI CULTURALI
Corso di Laurea Magistrale in Scienze Filosofiche

The Discriminating Power of Higher-Order Languages: A Process Algebraic Approach

Tesi di Laurea in Logica(1)(lm)

Relatore:
Prof.ssa Giovanna Corsi

Correlatore:
Prof. Davide Sangiorgi

Presentata da:
Valeria Vignudelli

Sessione II
Anno Accademico 2012/2013

Abstract

Questa tesi analizza differenti soluzioni volte a stabilire l'equivalenza di processi nondeterministici rappresentati da strutture matematiche note come Sistemi di Transizione Etichettati. La tesi è composta da due parti: nella prima parte introduciamo alcuni dei principali strumenti teorici e risultati emersi nella letteratura contemporanea, in particolare in ambito informatico; nella seconda parte della tesi proponiamo un nuovo approccio alle equivalenze, utilizzando un linguaggio di ordine superiore per testare il comportamento di un sistema.

La prima parte della tesi fornisce una panoramica delle principali equivalenze osservazionali su Sistemi di Transizione Etichettati, dalle meno discriminanti (equivalenze a tracce) alle più discriminanti (bisimilarità e isomorfismo). Introduciamo inoltre diverse logiche modali e mostriamo come queste permettano di ottenere caratterizzazioni alternative delle equivalenze discusse. I processi esaminati sono inizialmente processi puramente nondeterministici. Prendiamo quindi in considerazione Sistemi di Transizione Etichettati probabilistici, volti alla modellizzazione di processi in cui le transizioni di stato possono avere una specifica probabilità di essere effettuate.

Nella seconda parte della tesi l'equivalenza di due processi viene esaminata definendo un insieme di test tramite cui è possibile interagire con essi e determinare eventuali differenze nel loro comportamento. Il contributo originale di questa ricerca consiste nel considerare un linguaggio dei test che include operatori di ordine superiore. Applichiamo questo insieme di test sia a processi puramente nondeterministici che a processi con transizioni probabilistiche e dimostriamo due risultati principali.

Nel caso in cui i processi testati siano processi nondeterministici, caratterizziamo la indistinguibilità di due processi rispetto all'intera classe dei test con un'equivalenza nota nella letteratura come *ready simulation equivalence*. Il potere discriminante del linguaggio aumenta se applichiamo i test a processi probabilistici. In particolare, dimostriamo che l'equivalenza rispetto ai test coincide con la bisimilarità probabilistica.

Contents

List of Figures

Introduction

Labelled Transition Systems are widely used mathematical structures. In theoretical computer science, Labelled Transition Systems are a fundamental semantic tool describing the behavior of interactive systems. Formally, Labelled Transition Systems are relational structures, that is, sets equipped with one or more relations on its items defining the state-transitions that may occur at any stage of the computation. These relations are labelled, which means that they are indexed by names representing the actions the system can perform while interacting with the environment, such as input or output actions, or internal moves that are not observable from the outside.

What happens during the computations of an interactive system strictly depends on what happens in the environment it interacts with. This is the reason why nondeterministic processes are more suitable for describing the behavior of such a system than functions are. Thus, a state of a Labelled Transition System represents a stage that a process may reach after performing a given sequence of actions.

Modal Logics

A major field of study where Labelled Transition Systems play a central role is philosophical logic. Besides being possible models for classical first-order languages, these structures provide a mathematical semantics for modal logics, where they are known as Kripke frames.[1] As is well-known, the formal semantics of the concepts of necessity and possibility is defined over relational structures. Moreover, the interpretation of temporal logics (aimed at modeling temporarily qualified statements such as "F will always be true in the future" or " At least once in the past it was the case that F"), deontic logics (dealing with what is forbidden and what is allowed), epistemic logics (logics for reasoning about knowledge and beliefs) hinges on Kripke frames.

Modal logics have also been fruitfully applied in computer science. The expressiveness yet simplicity of modal languages make them a useful tool for checking properties of Labelled Transition Systems. For instance, basic modal languages are powerful enough to capture

[1] As pointed out in (Goldblatt 2006), it was over a period of three decades from the early 1930s that it became clear that modal logics can be interpreted on relational structures. Kripke frames are named after Saul Kripke (Kripke 1963).

safety properties and liveness properties of systems, that is, properties stating that nothing bad can happen (such as "it is never the case that the system does x") and something good will happen (such as "Now or then the system will execute x"), respectively. Hennessy-Milner Logic is such a modal language. Furthermore, Hennessy-Milner Logic allows us to alternatively characterize a well-known equivalence relation on nondeterministic processes: bisimilarity.[2]

Behavioral equivalences

It is not easy to understand what it means for two processes to have the same behavior. If we are only interested in the behavior of the systems, requiring that two systems are identical under renaming of the states (that is, requiring that they are isomorphic) is too strong a condition. At the same time, many equivalence relations defined in the literature are too underdiscriminating when applied to nondeterministic processes. Trace-based equivalences, which identify two processes by comparing the sequences of actions they can perform, are examples of such relations.

Bisimulation relations independently appeared both in modal logic and in computer science between the 1970s and the 1980s .[3] In his PhD thesis[4], Johan van Benthem addresses the problem of comparing the expressiveness of propositional modal languages and classical first-order languages. In this setting, Van Benthem defines *zigzag relations* on Kripke models and proves that modal formulas correspond to a fragment of classical first-order logic invariant under bisimulations. Contemporarily, the work of Robin Milner[5] and David Park[6] on the semantic of interactive systems gives rise to the notion of bisimulation, a relation on the states of a Labelled Transition System which coincides with Van Benthem's zigzag relations.[7]

Bisimulation relations induce an equivalence relation on processes, i.e. bisimilarity, which is taken to be a suitable notion of behavioral equivalence on Labelled Transition Systems. Bisimilarity also has a simple proof method: in order to prove that two processes are equivalent, we exhibit a relation containing the pair of processes and we verify that the relation is a bisimulation.[8]

[2] Hennessy-Milner Logic is named after Matthew Hennessy and Robin Milner (Hennessy and Milner 1985).

[3] See (Sangiorgi 2012b) for an historical analysis of the discovery of bisimulations.

[4] The thesis was discussed in 1974 and it is published in (van Benthem 1983).

[5] (Milner 1980, Milner 1989).

[6] (Park 1981).

[7] In the same period, bisimulations also appeared in set theory as relations for defining equivalences on non-well founded sets. See (Aczel 1988).

[8] This holds because bisimulations are coinductive relations. The definition of a coinductive set rests on the dual of the induction principle, i.e. the coinduction principle. The former defines a set by means of constructors; the latter defines a set by observing objects. See (Sangiorgi 2012a) for a fixed-point approach to coinduction and (Jacobs and Rutten 2012) for an algebraic approach.

Probabilistic processes

There are different possible extensions of Labelled Transition Systems. For instance, the definition of a Kripke model in modal logic rests on the enrichment of a Kripke frame obtained by associating to every state the set of propositional variables it satisfies. Probabilistic Labelled Transition Systems are another extension. Formally, a probabilistic Labelled Transition System is obtained by defining the transition relation as a relation from states to probability distributions on states. We can take a probabilistic process to be the refinement of a nondeterministic process determined after gaining new knowledge about the system, i.e. the knowledge concerning the probability a transition has of being performed.

Several models for describing probabilistic processes appear in the literature,[9] and their differences mainly lie in the interplay that is allowed between nondeterministic and probabilistic choices. In this work, we let probabilistic Labelled Transition Systems (or probabilistic processes) denote a general model where there are no limitations to this interplay. Equivalences and modal logics for probabilistic processes need to take into account the quantitative information that is now available in the structures they are applied to. Probabilistic bisimulations meet this requirement by considering not only the *possibility* but also the *probability* of performing a state-transition. Symmetrically, probabilistic modal languages do not simply check whether a formula is satisfied or not; rather, they verify the probability a formula has of being satisfied.

Process calculi and higher-order languages

Concurrency theory investigates the behavior of concurrent programs, that is, programs interacting with one another while running in parallel. As has emerged previously, processes (formalized by means of Labelled Transition Systems) are a suitable semantics for interactive systems. Process calculi (or process algebras) are a linguistic support for specifying concurrent programs.

The language of a process calculus includes operators for representing communications and the parallel composition of terms. A process calculus is a calculus since it is equipped with a syntax-driven semantics[10] which identifies the terms of the language with the states of a process and allows us to derive the transitions from a state by means of a set of rules. As a result, a Labelled Transition System is associated to any term of the process calculus, which in turn specifies a concurrent system.

These general features constitute the common basis of a large family of process calculi

[9] A comparison of a variety of probabilistic models is in (Sokolova and Vink 2004).

[10] This syntax-driven semantics is the *structural operational semantic* first introduced by Gordon D. Plotkin (Plotkin 2004b,Plotkin 2004a).

developed in the last forty years.[11] Well-known examples of process calculi are CCS (*Calculus of Communicating Systems*),[12] CSP (*Communicating Sequential Processes*)[13] and ACP (*Algebra of Communicating Processes*).[14] In process calculi such as CCS, the synchronization of atomic input-output actions emitted by processes running in parallel is the only form of interaction allowed between terms. The π-calculus[15] is a more expressive calculus, in that it models processes that interact through channels where they can communicate channel names to one another. Differently, in higher-order process calculi the communication between processes includes the exchange of terms of the languages.

Higher-order languages are meant to model a powerful feature of programs, i.e. the possibility of taking programs themselves as input. The pure λ-calculus is a paradigmatic example of a higher-order language: the variables of the λ-calculus range over the whole class of the λ-terms and the β-reduction allows a function to take any term of the language as argument. The pure λ-calculus is a calculus of functions, hence the determinism of its computations does not make it a suitable language for modeling concurrent processes.

Higher-order process calculi include HOπ, a π-calculus with higher-order communication, and the Kell Calculus,[16] a family of process calculi which further extends the π-calculus with both higher-order operators and constructs for modeling distributed systems.

Testing equivalences

We have introduced behavioral equivalences on processes, but we have not discussed a fundamental aspect in the definition of a behavioral equivalence yet: what does it mean to observe the behavior of a system? Which notion of observer are we assuming?

To address this problem, Robin Milner suggests a " button-pushing scenario" where the observer performs experiments on an interactive system (the agent) presented as a black box:

> An agent may be thought of as a black box, equipped with a button for each experiment. It also has a green light, which is lit iff the agent is proceeding without responding to experiment. To attempt an experiment e on agent p we apply continuous pressure to the e-button; if the button goes down (after some time) then p has accepted the experiment, and if the green light goes off without the button moving then p has rejected the experiment. While neither occurs (and if p can diverge then it is possible that neither will occur) we can conclude nothing.

[11] For a brief reconstruction of the origins and the evolution of process algebras we refer the reader to (Baeten 2005).

[12] (Milner 1980, Milner 1989).

[13] (Hoare 1978).

[14] (Bergstra and Klop 1984).

[15] (Sangiorgi and Walker 2001).

[16] (Schmitt and Stefani 2005).

(Milner 1981:26)

Thus, it is quite natural to try to characterize behavioral equivalences by means of experiments, or tests. In order to formalize this approach, one should rigorously define what a testing scenario is and what it means for two processes to be indistinguishable in the testing scenario.
Rocco De Nicola and Matthew Hennessy[17] first proposed a general setting for testing nondeterministic processes and defined three testing-based equivalences: *may*-equivalence, *must*-equivalence and *test*-equivalence. Samson Abramsky[18] applied these ideas to bisimilarity. In his testing scenario, the experimenter is allowed to test whether a process can or cannot perform an action, to make an unlimited number of copies of the process and to have a global perspective on the nondeterministic branches at any stage of the process. As a result, *test*-equivalence and bisimilarity collapse into one.

Outline of the thesis

This thesis analyzes various approaches to the behavioral equivalence of processes represented as Labelled Transition Systems. The thesis is divided into two parts: the first part (Chapter 1 and Chapter 2) aims at introducing some of the main results in the field, while in the second (Chapter 3) we propose a new approach to testing equivalences by considering a higher-order language of tests.

Chapter 1 is devoted to an overview of equivalence relations on nondeterministic processes, from the coarser ones (trace-based equivalences) to the finer ones (bisimilarity and isomorphism). Analogously to the characterization of bisimilarity through Hennessy-Milner logic, alternative characterizations of these equivalences on processes are given through a variety modal languages. Finally, we introduce the process calculus CCS.

Chapter 2 analyzes probabilistic Labelled Transition Systems and probabilistic bisimilarity. Reactive probabilistic processes are an interesting class of probabilistic processes since we can regard them as a first refinement of nondeterministic processes where internal choices are equipped with fixed probability values. We present both a logical and a testing-based characterization of bisimilarity on reactive probabilistic processes.

In Chapter 3 we investigate the equivalences that can be recovered by testing both the class of nondeterministic processes and the class of reactive probabilistic processes. The novelty of the approach put forward in this thesis lies in considering testing scenarios where contexts of the nondeterministic higher-order language HOL play the role of tests. HOL is inspired by the Kell Calculus and besides some of the usual CCS operators it

[17] (De Nicola and Hennessy 1984).

[18] (Abramsky 1987).

includes constructs for localities, passivation of localities, local communication and refusal of actions. These constructs allow us to define tests with two powerful features: testing the impossibility of performing an action and executing multiple tests on the process at any time of the computation, by copying it into different localities. We prove two main results.

If a test T in HOL is executed on a fully nondeterministic process P, we can observe whether P may pass T (i.e. there exists a successful execution of the test) or whether P must pass T (i.e. there are no unsuccessful executions). Two nondeterministic processes P and Q are *may*-equivalent (respectively: *must*-equivalent) if P and Q may (must) pass exactly the same tests. We prove that in this testing scenario ready-simulation equivalence, viz. a slightly coarser relation than bisimilarity, coincides with *may*-equivalence, which in turn implies *must*-equivalence.

Finally, HOL-contexts are employed to test reactive probabilistic processes. In this case, it is possible to observe whether a process may, or must, pass a test with a specific probability, and the testing equivalences require these values to coincide. Knowing the probabilities of passing a test increases the discriminating power of HOL-contexts. In particular, we prove that in this testing scenario there is a collapse of equivalences: probabilistic bisimilarity coincides with both *may*-equivalence and *must*-equivalence.

Chapter 1

Behavioral equivalences on nondeterministic processes

In this chapter we introduce Labelled Transition Systems, namely relational structures representing nondeterministic processes. Bisimilarity is a suitable notion of behavioral equivalence for this semantic (Aceto et al. 2007, Sangiorgi 2012a): Section 1.2 is devoted to introducing bisimulations and the alternative characterizations of bisimilarity through inductively defined relations and Hennessy-Milner Logic. Section 1.3 justifies the pivotal role played by bisimilarity. We compare bisimilarity to several equivalence relations on nondeterministic processes and we argue that each of these relations either equates too many processes or distinguishes structures that an observer would consider the same. Finally, Section 1.4 presents the Calculus of Communicating Systems, viz. a language for describing concurrent programs and their interactions.

1.1 Labelled Transition Systems

A Labelled Transition System is a tuple which consists of a set of states, a set of atomic actions, or labels, and a labelled transition relation on the set of states. To put it differently, Labelled Transition Systems are labelled directed graphs.
These structures are widely used in computer science as abstract models of concurrent systems.[19] We can look at the states of a Labelled Transition System as nondeterministic processes evolving into other processes by means of actions representing their communications or interactions with the environment (Milner 1989).

Definition 1.1. A *Labelled Transition System (LTS)* is a tuple $\mathcal{T} = \langle St, \mathscr{A}, \longrightarrow \rangle$ where:

[19] Labelled Transition System are pervasive mathematical structures. For instance, in modal logic they are multisorted Kripke models whose set of propositional variables is empty; see (Blackburn, de Rijke, and Venema 2001).

- St is a non-empty set of states or processes,

- $\mathscr{A}$ is a set of atomic actions,

- $\longrightarrow \subseteq St \times \mathscr{A} \times St$ is a labelled transition relation on the set of states.

In what follows, we use $P, Q, R, S...$ (and their indexed variants $P_1, P_2 \ldots, R_1, R_2, \ldots$) to range over processes and μ (and its indexed variants $\mu_1, \mu_2 \ldots$) to range over $\mathscr{A}$, whose elements are denoted by $a, b, c, \ldots$.

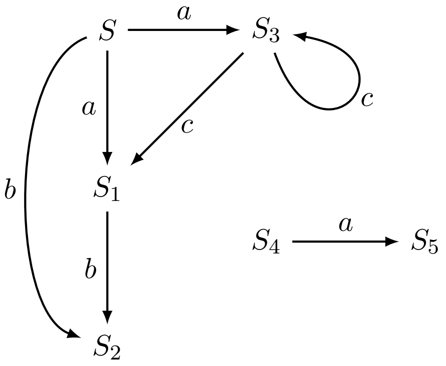

Figure 1.1: An LTS.

For instance, Figure 1.5 is the graphical representation of the LTS $\mathcal{T} = \langle St, \mathscr{A}, \longrightarrow \rangle$, where:

$$\begin{aligned} St =& \{S, S_1, S_2, S_3, S_4, S_5\} \\ \mathscr{A} =& \{a, b, c\} \\ \longrightarrow =& \{(S, a, S_1), (S, b, S_2), (S_1, b, S_2), (S, a, S_3), (S_3, c, S_3), (S_3, c, S_1), (S_4, a, S_5)\}. \end{aligned}$$

Given any LTS $\mathcal{T} = \langle St, \mathscr{A}, \longrightarrow \rangle$, we write:

$$\begin{array}{ll} P \xrightarrow{\mu} P' & \text{if } (P, \mu, P') \in \longrightarrow \\ P \overset{\mu}{\not\longrightarrow} P' & \text{if } (P, \mu, P') \notin \longrightarrow \\ P \overset{\mu}{\not\longrightarrow} & \text{if } P \overset{\mu}{\not\longrightarrow} P' \text{ for all processes } P' \\ P \not\longrightarrow & \text{if } P \overset{\mu}{\not\longrightarrow} \text{ for all atomic actions } \mu. \end{array}$$

Definition 1.2. Let P be a process in $\mathcal{T} = \langle St, \mathscr{A}, \longrightarrow \rangle$. A process P' in $\mathcal{T}$ is *reachable from* P if there is an $n \in \mathbb{N}$ such that:

$$P = P_0 \xrightarrow{\mu_1} P_1 \xrightarrow{\mu_2} \ldots \xrightarrow{\mu_n} P_n$$

for some processes $P_0, P_1, \ldots, P_n$ and for some atomic actions $\mu_1, \mu_2, \ldots, \mu_n$.
The LTS reachable from P is the LTS $\mathcal{T}_P = \langle St_P, \mathscr{A}_P, \longrightarrow_P \rangle$ defined as follows:

- $St_P \subseteq St$ is the set of states reachable from P, i.e. the set:

$$\{P_n | \, P \xrightarrow{\mu_1} P_1 \xrightarrow{\mu_2} \ldots \xrightarrow{\mu_n} P_n, \text{ for some } n \in \mathbb{N}, P_1, \ldots, P_{n-1}, \mu_1, \ldots, \mu_n\},$$

- $\mathscr{A}_P = \{\mu | \mu \in \mathscr{A} \text{ and } P' \xrightarrow{\mu} \text{ for some } P' \in St_P\}$,
- $\longrightarrow_P$ is the restriction of $\longrightarrow$ to St_P and $\mathscr{A}_P$.

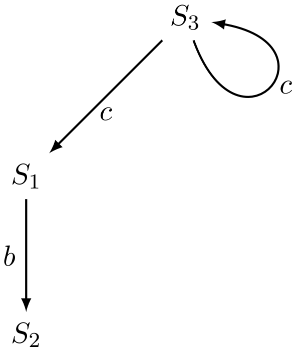

Figure 1.2: The LTS reachable from the process S_3 in Figure 1.1.

An LTS $\mathcal{T} = \langle St, \mathscr{A}, \longrightarrow \rangle$ is:

- *finite* if its set of states St is finite and it has no loops, that is, for every $P, P' \in St$ if P' is reachable from P then $P \neq P'$,
- *finite-state* if its set of states is finite,
- *finitely branching* if for every state P, the set of P's derivatives (i.e. the set $\{P' | \, P \xrightarrow{\mu} P' \text{ for some } \mu\}$) is finite,
- *image-finite* if for every state P and for every μ, the set of P's μ-derivatives (i.e. the set $\{P' | \, P \xrightarrow{\mu} P'\}$) is finite.

We adapt these definitions to a specific process P by considering the LTS $\mathcal{T}_P = \langle St_P, \mathscr{A}_P, \longrightarrow_P \rangle$ reachable from P in place of the LTS $\mathcal{T}$.

1.2 Bisimilarity

Bisimulation relations were independently defined in three different research fields. In the setting of modal logic, Johan van Benthem introduced them as *zigzag relations* in his

1976 PhD Thesis, printed in (van Benthem 1983). In (Milner 1989) Robin Milner defined bisimilarity, i.e. the equivalence relation induced by bisimulation relations, on Labelled Transition Systems, in the setting of concurrency theory. Finally, Marco Forti, Furio Honsell and Peter Aczel (Aczel 1988) proposed the notion of bisimilarity in the setting of non-well-founded set theory.[20]

Bisimulations have proved to be a successful tool to capture and to demonstrate the equivalence of processes: in order to prove that two processes P and Q have the same behaviour we exhibit a relation including the pair (P, Q) and satisfying the definition of bisimulation.[21]

Definition 1.3. Let $\mathcal{T} = \langle St, \mathscr{A}, \longrightarrow \rangle$ be an LTS. A relation $\mathcal{R} \subseteq St \times St$ is a *bisimulation* if $P \, \mathcal{R} \, Q$ implies:

- for every $\mu \in \mathscr{A}$, $P' \in St$, if $P \xrightarrow{\mu} P'$ then $Q \xrightarrow{\mu} Q'$ and $P' \, \mathcal{R} \, Q'$, for some $Q' \in St$,
- for every $\mu \in \mathscr{A}$, $Q' \in St$, if $Q \xrightarrow{\mu} Q'$ then $P \xrightarrow{\mu} P'$ and $P' \, \mathcal{R} \, Q'$, for some $P' \in St$.

We say that P and Q are *bisimilar* ($P \sim Q$) if there exists a bisimulation $\mathcal{R}$ such that $P \, \mathcal{R} \, Q$. Hence, the *bisimilarity* relation $\sim$ is the union of all the bisimulations on an LTS.

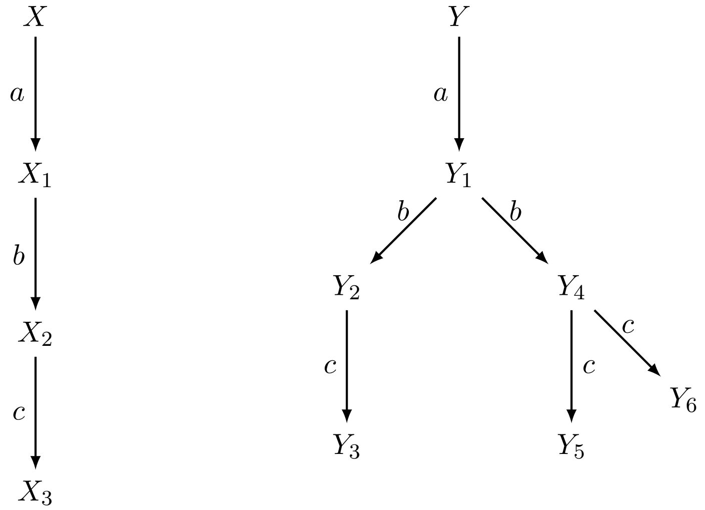

Figure 1.3: Bisimilar processes.

For instance, let us consider the LTS depicted in Figure 1.3. The relation:

$$\{(X, Y), (X_1, Y_1), (X_2, Y_2), (X_3, Y_3), (X_2, Y_4), (X_3, Y_5), (X_3, Y_6)\}$$

[20] We refer the reader to (Sangiorgi 2012b) for a detailed reconstruction of the origins of bisimulations.

[21] The validity of the bisimulation proof method rests on the dual of the induction principle: the coinduction principle. See (Sangiorgi 2012a).

is a bisimulation, thus X and Y are bisimilar.
The reason why we do not want to distinguish the two processes from a behavioral point of view is that the different structure they have does not affect their behavior. The process X only can do a, then b, then c and stop; the same holds for the process Y.

Theorem 1.4. *Let $\mathcal{T} = \langle St, \mathscr{A}, \longrightarrow\rangle$ be an LTS.*

1. *$\sim$ is an equivalence relation,*
2. *$\sim$ is the largest bisimulation on $\mathcal{T}$.*

Proof.

1. We prove that $\sim$ is reflexive, symmetric and transitive.
   - Let $\mathcal{I}$ be the identity relation on the processes in $\mathcal{T}$. $\mathcal{I}$ is a bisimulation, for $P \xrightarrow{\mu} P'$ implies $P \xrightarrow{\mu} P'$ and $(P', P') \in \mathcal{I}$, and vice versa. Then $P \sim P$ holds for every process P.
   - If $\mathcal{R}$ is a bisimulation then by switching the two conditions in the definition of bisimulation we obtain that $\mathcal{R}^{-1} = \{(P, Q) | \, Q \, \mathcal{R} \, P\}$ is a bisimulation too. Therefore, $P \sim Q$ implies that $Q \sim P$.
   - Suppose that $P \sim P_1 \sim P_2$. Then there are two bisimulation relations $\mathcal{R}_1$ and $\mathcal{R}_2$ such that $P \, \mathcal{R}_1 P_1$ and $P_1 \, \mathcal{R}_2 P_2$. We prove that $\mathcal{R}_3 = \mathcal{R}_1 \circ \mathcal{R}_2 = \{(P, P_2) |$ there exists a P_1 such that $P \, \mathcal{R}_1 P_1$ and $P_1 \, \mathcal{R}_2 P_2\}$ is a bisimulation relation, which in turn implies that $P \sim P_2$.
     Let $P \, \mathcal{R}_3 P_2$ and let $P \xrightarrow{\mu} P'$. Then $P_1 \xrightarrow{\mu} P_1'$ and $P' \, \mathcal{R}_1 P_1'$, for some P_1 such that $P \, \mathcal{R}_1 P_1 \, \mathcal{R}_2 P_2$ and for some P_1'. We derive from $P_1 \, \mathcal{R}_2 P_2$ that there is a P_2' such that $P_2 \xrightarrow{\mu} P_2'$ and $P_1' \, \mathcal{R}_2 P_2'$. It follows from the definition of $\mathcal{R}_3$ that $P' \, \mathcal{R}_3 P_2'$. Symmetrically, we have that $P \, \mathcal{R}_3 P_2$ and $P_2 \xrightarrow{\mu} P_2'$ implies that $P \xrightarrow{\mu} P'$ for some P' such that $P' \, \mathcal{R}_3 P_2'$. Thus, $\sim$ is a transitive relation.
2. Let $P \xrightarrow{\mu} P'$ and $P \sim Q$. Then there is a Q' such that $Q \xrightarrow{\mu} Q'$ and $P' \, \mathcal{R} \, Q'$, where $\mathcal{R}$ is a bisimulation relating P and Q. Hence, $P' \sim Q'$. The second condition in the definition of bisimulation is analogously satisfied by $\sim$, which thereby is a bisimulation.
   Finally, we have that $\sim$ is the largest bisimulation on $\mathcal{T}$, since by definition all bisimulation relations are included in $\sim$.

□

Theorem 1.5 provides a first alternative characterization of bisimilarity.

Theorem 1.5. *Let $\mathcal{T} = \langle St, \mathscr{A}, \longrightarrow\rangle$ be an LTS. $P \sim Q$ if and only if:*

- *for every $\mu \in \mathscr{A}$, $P' \in St$, if $P \xrightarrow{\mu} P'$ then $Q \xrightarrow{\mu} Q'$ and $P' \sim Q'$, for some $Q' \in St$,*
- *for every $\mu \in \mathscr{A}$, $Q' \in St$, if $Q \xrightarrow{\mu} Q'$ then $P \xrightarrow{\mu} P'$ and $P' \sim Q'$, for some $P' \in St$.*

Proof. Define the relation $\sim'$ such that $P \sim' Q$ if and only if:

- for every $\mu \in \mathscr{A}$, $P' \in St$, if $P \xrightarrow{\mu} P'$ then $Q \xrightarrow{\mu} Q'$ and $P' \sim Q'$, for some $Q' \in St$,
- for every $\mu \in \mathscr{A}$, $Q' \in St$, if $Q \xrightarrow{\mu} Q'$ then $P \xrightarrow{\mu} P'$ and $P' \sim Q'$, for some $P' \in St$.

We prove that $\sim=\sim'$.
By Theorem 1.4, $\sim$ is a bisimulation. If $P \sim Q$ then the two conditions above hold, which implies that $P \sim' Q$. Therefore, bisimilarity is included in $\sim'$.
As for the other direction, we show that $\sim'$ is a bisimulation. If $P \sim' Q$ and $P \xrightarrow{\mu} P'$ then $Q \xrightarrow{\mu} Q'$ and $P' \sim Q'$, for some $Q' \in St$. We proved above that $\sim\subseteq\sim'$, so the first condition in the definition of bisimulation is satisfied. Symmetrically, if $Q \xrightarrow{\mu} Q'$ then $P \xrightarrow{\mu} P'$ and $P' \sim Q'$, which implies that $P' \sim' Q'$. □

1.2.1 Approximants of bisimilarity

Under certain conditions it is possible to recover bisimilarity as the intersection of a decreasing chain of inductively defined relations[22]. Each of these relations is indexed by a natural number n, representing the relative degree of approximation of bisimilarity: the relation $\sim_n$ is the set of those processes which turn out to be bisimilar when only the states reachable from a process in n steps are checked.

Definition 1.6. Let $\mathcal{T} = \langle St, \mathscr{A}, \longrightarrow \rangle$ be an *LTS*. We define by induction on $n \in \mathbb{N}$ the sequence of relation $\sim_0, \sim_1, \ldots, \sim_n, \ldots$:

$P \sim_0 Q$ always holds,

$P \sim_{n+1} Q$ if and only if:

- for every $\mu \in \mathscr{A}$, $P' \in St$, if $P \xrightarrow{\mu} P'$ then $Q \xrightarrow{\mu} Q'$ and $P' \sim_n Q'$, for some $Q' \in St$,
- for every $\mu \in \mathscr{A}$, $Q' \in St$, if $Q \xrightarrow{\mu} Q'$ then $P \xrightarrow{\mu} P'$ and $P' \sim_n Q'$, for some $P' \in St$.

[22] The definition of bisimilarity through a stratification of inductively defined relations was given in (Hennessy and Milner 1985), under the name of *observational equivalence.*

If $P \sim_n Q$ we say that P and Q are *n-bisimilar.* Let

$$\sim_\omega = \bigcap_{n \in \mathbb{N}} \sim_n .$$

If $P \sim_\omega Q$ we say that P and Q are *ω-bisimilar.*

Theorem 1.7. *Let $\mathcal{T} = \langle St, \mathscr{A}, \longrightarrow \rangle$ be an LTS. If $P \sim Q$ then $P \sim_n Q$ for all $n \in \mathbb{N}$.*

Proof. By induction on n.

$(n = 0)$ The result follows from the fact that $P \sim_0 Q$ always holds.

$(n + 1)$ Bisimilarity is itself a bisimulation (Theorem 1.4), so $P \sim Q$ implies that

- for every $\mu \in \mathscr{A}$, $P' \in St$, if $P \xrightarrow{\mu} P'$ then $Q \xrightarrow{\mu} Q'$ and $P' \sim Q'$, for some $Q' \in St$,
- for every $\mu \in \mathscr{A}$, $Q' \in St$, if $Q \xrightarrow{\mu} Q'$ then $P \xrightarrow{\mu} P'$ and $P' \sim Q'$, for some $P' \in St$.

By the inductive hypothesis, $P' \sim Q'$ implies $P' \sim_n Q'$ and the result follows from the definition of $\sim_{n+1}$.

□

Corollary 1.8. *Let $\mathcal{T} = \langle St, \mathscr{A}, \longrightarrow \rangle$ be an LTS. If $P \sim Q$ then $P \sim_\omega Q$.*

Proof. By definition, ω-bisimilarity is the intersection of all n-bisimilarities, for $n \in \mathbb{N}$. Then, by Theorem 1.7, ω-bisimilarity is coarser than bisimilarity.

□

The converse of Corollary 1.8 does not hold on arbitrary LTSs. Consider the processes P and Q in Figure 1.4. We prove by induction on n that $P \sim_n Q$ for all $n \in \mathbb{N}$, which implies that $P \sim_\omega Q$

The base case always holds, so let us consider the inductive step. If $P \xrightarrow{a} P'$ then $P' = P^1_k$ for some k and there exists a $Q' = Q^1_k$ such that $Q \xrightarrow{a} Q'$ and $P' \sim Q'$, being the relation $\{(P^h_k, Q^h_k) | \, 1 \leq h \leq k\}$ a bisimulation. It follows from Theorem 1.7 that $P' \sim_n Q'$. The same holds whenever $Q \xrightarrow{a} Q'$ and $Q' = Q^1_k$. Now consider the case when $Q \xrightarrow{a} Q_\omega$. The process P reaches P^1_{n+1} through an a-labelled transition and we have that $(P^{n+1-h}_{n+1}, Q_\omega) \in \sim_h$ for every h from 0 to n. We show this by induction on h. The base case $(P^{n+1}_{n+1}, Q_\omega) \in \sim_0$ always holds. For the inductive step, suppose that $(P^{n+1-h}_{n+1}, Q_\omega) \in \sim_h$, where $h < n$. If $P^{n+1-(h+1)}_{n+1} \xrightarrow{\mu} P'$ then $\mu = a$ and $P' = P^{n+1-h}_{n+1}$. The process Q_ω matches this transition by doing $Q_\omega \xrightarrow{a} Q_\omega$, where $(P^{n+1-h}_{n+1}, Q_\omega) \in \sim_h$. Symmetrically, if Q_ω loops over itself through an a-labelled transition then $P^{n+1-(h+1)}_{n+1} \xrightarrow{\mu} P^{n+1-h}_{n+1}$, where

$(P_{n+1}^{n+1-h}, Q_\omega) \in \sim_h$. Hence, P_{n+1}^1 and Q_ω are n-bisimilar and the process P is therefore $n+1$-bisimilar to Q.
Suppose now that $P \sim Q$. Then there is a process P' such that $P \xrightarrow{a} P'$ and $P' \sim Q_\omega$, that is, there is an n such that $P_n^1 \sim Q_\omega$. As a consequence, for every h such that $1 \leq h \leq n$ we have that $P_n^h \sim Q_\omega$, which is absurd because $P_n^n \not\longrightarrow$ and $Q_\omega \xrightarrow{a}$. Thus, P and Q are not bisimilar.

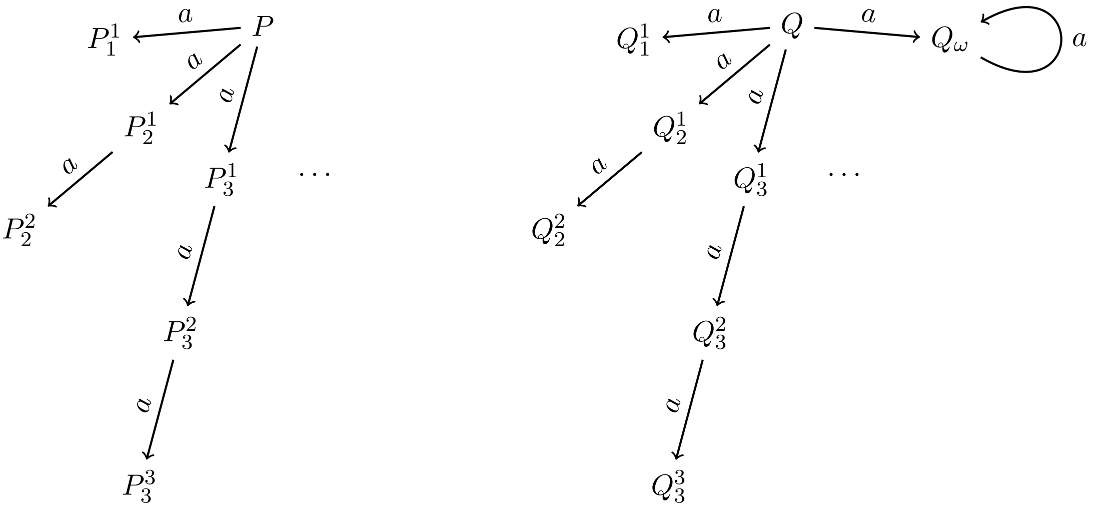

Figure 1.4: $\sim_\omega \not\subseteq \sim$

We will prove in the following section that a sufficient condition for the converse of Theorem 1.7 to hold is image-finiteness.

1.2.2 Hennessy-Milner Logic

Hennessy-Milner Logic is a modal language introduced in (Hennessy and Milner 1985). HML allows us to derive an alternative characterization of bisimilarity on image-finite processes: if P and Q are image-finite processes, then P and Q are bisimilar if and only if they satisfy all and only the same HML-formulas.

Definition 1.9. Let $\mathscr{A}$ be a set of atomic actions. The formulas of *Hennessy-Milner Logic*[23] *(HML)* on $\mathscr{A}$ are defined as follows:

$$F ::= \top \;\Big|\; \neg F \;\Big|\; F_1 \wedge F_2 \;\Big|\; \langle\, \mu \,\rangle F$$

[23] We present a modal logic equipped with the negation operator, as originally defined in (Hennessy and Milner 1985). However, the negation operator can be eliminated if the disjunctive formula $F_1 \vee F_2$, logically equivalent to the HML- formula $\neg(\neg F_1 \wedge \neg F_2)$, and the modality $[\,\mu\,]F$, logically equivalent to the HML-formula $\neg\langle\,\mu\,\rangle\neg F$, are added to HML. All the results presented in this section remain valid if we consider this alternative definition of HML (Aceto et al. 2007).

We define by structural induction on HML-formulas when $P \models F$ ("the HML-formula F is true at the process P" or "the process P satisfies the HML-formula F"):

$P \models \top$	always	
$P \models \neg F$	iff	$P \not\models F$
$P \models F_1 \wedge F_2$	iff	$P \models F_1$ and $P \models F_2$
$P \models \langle \mu \rangle F$	iff	there is a P' such that $P \xrightarrow{\mu} P'$ and $P' \models F$.

$P \equiv^{HML} Q$ if and only if for every HML-formula F, $P \models F$ if and only if $Q \models F$.

In order to relate the approximants of bisimilarity introduced in the previous section to the satisfaction of HML-formulas, we associate to every HML-formula its modal degree.

Definition 1.10. The *modal degree* ($md(F)$) of an HML-formula F is defined by structural induction on F:

$$\begin{aligned} md(\top) &= 0 \\ md(\neg F_1) &= md(F_1) \\ md(F_1 \wedge F_2) &= max\{md(F_1), md(F_2)\} \\ md(\langle \mu \rangle F_1) &= 1 + md(F_1). \end{aligned}$$

$P \equiv_n^{HML} Q$ if and only if for every HML-formula F such that $md(F) \leq n$, $P \models F$ if and only if $Q \models F$.

Intuitively, the modal degree of an HML-formula F corresponds to the number of transitions from a process that F can "see". The following theorem supports this intuition.

Theorem 1.11. *Let $\mathcal{T} = \langle St, \mathscr{A}, \longrightarrow \rangle$ be an LTS. For every $n \in \mathbb{N}$, if $P \sim_n Q$ then $P \equiv_n^{HML} Q$.*

Proof. We prove the result by induction on n. It is easy to check that $md(F) \leq 0$ implies that F is logically equivalent to either $\top$ or $\neg\top$. Therefore, $P \equiv_0^{HML} Q$ always holds and the base case follows.
Suppose now that $P \sim_{n+1} Q$. We prove by structural induction on F that $md(F) = n+1$ implies that $P \models F$ if and only if $Q \models F$.

($F = \top$) $md(\top) = 0 \neq n+1$, so this case vacuously hold.

($F = \neg F'$) Let $md(F) = n+1$. Then $md(F') = md(\neg F') = n+1$. We have that $P \models F$ if and only if $P \not\models F'$ if and only if (by the inductive hypothesis on F') $Q \not\models F'$ if and only if $Q \models F$.

($F = F_1 \wedge F_2$) Analogously to the previous case, $md(F) = n + 1$ implies that $md(F_1) = md(F_2) = md(F_1 \wedge F_2) = n + 1$. Then $P \models F_1 \wedge F_2$ if and only if $P \models F_1$ and $P \models F_2$ if and only if (by the inductive hypothesis on F_1 and F_2) $Q \models F_1$ and $Q \models F_2$ if and only if $Q \models F_1 \wedge F_2$.

($F = \langle\mu\rangle F'$) Let $md(F) = n + 1$. Then $md(F') = md(F) - 1 = n$. If $P \models \langle\mu\rangle F'$ then there exists a P' such that $P \xrightarrow{\mu} P'$ and $P' \models F$. It follows from the hypothesis that P and Q are $n+1$-bisimilar that $Q \xrightarrow{\mu} Q'$, for some Q' such that $P' \sim_n Q'$. By the inductive hypothesis on n we have that $P' \equiv_n^{HML} Q$. Therefore, $Q' \models F'$, which implies that $Q \models \langle\mu\rangle F'$.
Symmetrically, $Q \models \langle\mu\rangle F'$ implies $P \models \langle\mu\rangle F'$.

□

It follows from Theorem 1.11 that ω-bisimilar processes satisfy all and only the same HML-formulas. In the previous section we proved that the processes P and Q in Figure 1.4 are ω-bisimilar but not bisimilar processes. As a consequence of Theorem 1.11, there does not exist an HML-formula which can distinguish P and Q .
A specific feature of the processes in Figure 1.4 is that they can perform infinitely many different a-labelled transitions. If we restrict our attention to image-finite LTSs, the relations $\sim$, $\sim_\omega$ and $\equiv^{HML}$ coincide.

Theorem 1.12. *Let $\mathcal{T} = \langle St, \mathscr{A}, \longrightarrow\rangle$ be an image-finite LTS. $P \sim Q$ if and only if $P \equiv^{HML} Q$.*

Proof.

($\Rightarrow$) By Theorem 1.8, bisimilarity implies ω-bisimilarity. If $P \sim_\omega Q$ then $P \sim_n Q$ for all $n \in \mathbb{N}$. It follows from Theorem 1.11 that $P \equiv_n^{HML} Q$ for all $n \in \mathbb{N}$, which implies that $P \equiv^{HML} Q$.

($\Leftarrow$) We prove that $\equiv^{HML}$ is a bisimulation, which in turn implies the result.
Let $P \equiv^{HML} Q$ and $P \xrightarrow{\mu} P'$. Then $\langle\mu\rangle\top$ is true at P and it follows from the hypothesis of equivalence with respect to HML-formulas that $Q \models \langle\mu\rangle\top$. Hence, there is at least a process Q' such that $Q \xrightarrow{\mu} Q'$. Let $\{Q_1, \ldots, Q_n\}$ be the set Q's μ-derivatives, i.e. the set of processes reached by Q through a μ-labelled transition. By the image-finiteness hypothesis, this set is finite. We prove by contradiction that there is an i such that $1 \leq i \leq n$ and $P' \equiv^{HML} Q_i$. Suppose not. Then there exists a formula F_i such that $P' \models F_i$ and $Q_i \models \neg F_i$ for every $i \in \{1, \ldots, n\}$. Therefore, $P \models \langle\mu\rangle(F_1 \wedge \ldots \wedge F_n)$, which implies $Q \models \langle\mu\rangle(F_1 \wedge \ldots \wedge F_n)$, by the hypothesis that $P \equiv^{HML} Q$. However, Q' is a Q's μ-derivative if and only if $Q' = Q_i$ for some $i \in \{1, \ldots, n\}$, and for every i we have that $Q_i \not\models F_i$. As a consequence,

$Q' \not\models (F_1 \wedge \ldots \wedge F_n)$ for every Q' such that $Q \xrightarrow{\mu} Q'$. This contradicts the assumption that $Q \models \langle\, \mu \,\rangle(F_1 \wedge \ldots \wedge F_n)$.

The proof of the second condition in the definition of bisimulation (i.e. $P \equiv^{HML} Q$ and $Q \xrightarrow{\mu} Q'$ implies $P \xrightarrow{\mu} P'$, for some P' such that $P' \equiv^{HML} Q'$) is symmetrical.

□

1.3 Why bisimilarity?

We presented bisimilarity as a suitable definition of behavioral equivalence on processes. In order to justify this statement, we present a spectrum of equivalences[24] , from the coarser ones to the finer ones.

We start from trace-based equivalences, which compare two processes by considering the sequences of actions they can perform, and we end with isomorphism between the LTSs reachable from the processes. For each of these relations, we define a logic which characterizes it and we prove the analogous of the Hennessy-Milner Theorem (Theorem 1.12) for that specific case.

1.3.1 Trace equivalence

Trace equivalence was first introduced in (Hoare 1980). Every process has a set of traces, i.e. the set of sequences of atomic actions it can perform, and two processes are equivalent if their sets of traces coincide.

Definition 1.13. Let $\mathcal{T} = \langle St, \mathscr{A}, \longrightarrow \rangle$ be an LTS.

- For any $n \in \mathbb{N}$, a sequence $\mu_1, \ldots, \mu_n$ of atomic actions is a *trace* from P if there are processes $P_1, \ldots, P_n$ such that:

$$P \xrightarrow{\mu_1} P_1 \xrightarrow{\mu_2} P_2 \xrightarrow{\mu_3} \ldots \xrightarrow{\mu_n} P_n.$$

  We let ϵ denote the empty trace and $Tr(P) = \{\sigma |\ \sigma \text{ is a trace from } P\}$.

- P and Q are *trace equivalent* ($P \equiv_{Tr} Q$) if $Tr(P) = Tr(Q)$.

The modal logic for trace equivalence (Trace Logic) is the subset of Hennessy-Milner Logic whose only operators are the top operator $\top$ and the modal operator $\langle\, \mu \,\rangle$.

Definition 1.14. Let $\mathscr{A}$ be a set of atomic actions. The formulas of *Trace Logic (TL)* on $\mathscr{A}$ are defined as follows:

$$F ::= \top \;\Big|\; \langle\, \mu \,\rangle F$$

[24] This section draws on (van Glabbeek 2001), which studies a more detailed spectrum than the one here considered.

The satisfaction of TL-formulas follows the definition of truth of an HML-formula at a state (Definiton 1.9):

$$\begin{array}{lll} P \models \top & \text{always} & \\ P \models \langle\,\mu\,\rangle F & \text{iff} & \text{there is a } P' \text{ such that } P \xrightarrow{\mu} P' \text{ and } P' \models F. \end{array}$$

$P \equiv^{TL} Q$ if and only if for every TL-formula F, $P \models F$ if and only if $Q \models F$.

Lemma 1.15. *Let $\mathcal{T} = \langle St, \mathscr{A}, \longrightarrow\rangle$ be an LTS. For every $n \in \mathbb{N}$, for every sequence $\mu_1, \ldots, \mu_n$ of atomic actions and for every process P,*

$$(\mu_1, \ldots, \mu_n) \in Tr(P) \textit{ if and only if } P \models \langle\,\mu_1\,\rangle \ldots \langle\,\mu_n\,\rangle \top.$$

Proof. By induction on the length n of the sequence $\mu_1, \ldots, \mu_n$.

$(n = 0)$ The empty trace ϵ always belongs to $Tr(P)$ and $\top$ is always true at P.

$(n+1)$ If $(\mu_1, \ldots, \mu_{n+1})$ is a trace of P then $P \xrightarrow{\mu_1} P_1 \xrightarrow{\mu_2} \ldots \xrightarrow{\mu_{n+1}} P_{n+1}$, for some $P_1, \ldots, P_{n+1}$. By the inductive hypothesis, $P_1 \xrightarrow{\mu_2} \ldots \xrightarrow{\mu_{n+1}} P_{n+1}$ if and only if $P_1 \models \langle\,\mu_2\,\rangle \ldots \langle\,\mu_{n+1}\,\rangle \top$. It follows from the fact that $P \xrightarrow{\mu_1} P_1$ that $P \models \langle\,\mu_1\,\rangle \ldots \langle\,\mu_{n+1}\,\rangle \top$. As for the other direction, suppose that $P \models \langle\,\mu_1\,\rangle \ldots \langle\,\mu_{n+1}\,\rangle \top$. Then there is a P_1 such that $P \xrightarrow{\mu_1} P_1$ and $P_1 \models \langle\,\mu_2\,\rangle \ldots \langle\,\mu_{n+1}\,\rangle \top$, which implies $P_1 \xrightarrow{\mu_2} \ldots \xrightarrow{\mu_{n+1}} P_{n+1}$ by the hypothesis of induction. Therefore, $(\mu_1, \ldots, \mu_{n+1}) \in Tr(P)$.

□

Given a set $\mathscr{A}$ of atomic actions, it is immediate to check that F is a TL-formula on $\mathscr{A}$ if and only if $F = \langle\,\mu_1\,\rangle \ldots \langle\,\mu_n\,\rangle \top$, for some $n \in \mathbb{N}$ and for some $\mu_1, \ldots, \mu_n$ in $\mathscr{A}$. This allows us to derive from Lemma 1.15 that Trace Logic characterizes trace equivalence.

Theorem 1.16. *Let $\mathcal{T} = \langle St, \mathscr{A}, \longrightarrow\rangle$ be an LTS. $P \equiv_{Tr} Q$ if and only if $P \equiv^{TL} Q$.*

Proof. For all $n \in \mathbb{N}$ and for all traces $\mu_1, \ldots, \mu_n$,

$$\begin{array}{lll} P \equiv_{Tr} Q & \text{iff} & (\mu_1, \ldots, \mu_n) \in Tr(P) \text{ iff } (\mu_1, \ldots, \mu_n) \in Tr(Q) \\ & \text{iff} & P \models \langle\,\mu_1\,\rangle \ldots \langle\,\mu_n\,\rangle \top \text{ iff } Q \models \langle\,\mu_1\,\rangle \ldots \langle\,\mu_n\,\rangle \top \qquad \text{(by Lemma 1.15)} \\ & \text{iff} & P \equiv^{TL} Q. \end{array}$$

□

The processes P and Q in Figure 1.5 are trace equivalent: $Tr(P) = \{a, ab\} = Tr(Q)$. However, P and Q should be differentiated from a behavioral point of view. A b-labelled action is available every time that P performs an a-labelled action; in contrast, Q does a and reaches Q_1, where b cannot be performed anymore.

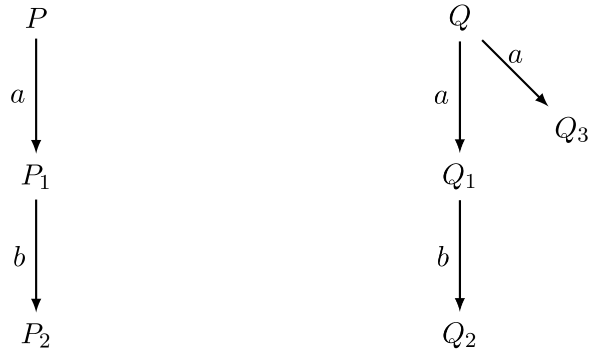

Figure 1.5: Trace equivalent processes.

1.3.2 Completed trace equivalence

Completed trace equivalence can observe whether a trace ends with a deadlock state such as Q_3, a state refusing all actions. Thus, completed trace equivalence discriminates the processes P and Q in Figure 1.5.

Definition 1.17. Let $\mathcal{T} = \langle St, \mathscr{A}, \longrightarrow\rangle$ be an LTS.

- for any $n \in \mathbb{N}$, a trace $\mu_1, \dots, \mu_n$ from P is a *completed trace* from P if there are processes $P_1, \dots, P_n$ such that:

  $$P \xrightarrow{\mu_1} P_1 \xrightarrow{\mu_2} P_2 \xrightarrow{\mu_3} \dots \xrightarrow{\mu_n} P_n \text{ and } P_n \not\longrightarrow .$$

  We let $CTr(P) = \{\sigma |\ \sigma \text{ is a complete trace from } P\}$.

- P and Q are *completed trace equivalent* ($P \equiv_{CTr} Q$) if $Tr(P) = Tr(Q)$ and $CTr(P) = CTr(Q)$.

The completed traces of the processes P and Q in Figure 1.5 are $\{ab\}$ and $\{a, ab\}$, respectively. Therefore, $P \not\equiv_{CTr} Q$.

Conversely, it directly follows from Definition 1.17 that completed trace equivalence is included in trace equivalence:

Theorem 1.18. *Let $\mathcal{T} = \langle St, \mathscr{A}, \longrightarrow\rangle$ be an LTS. If $P \equiv_{CTr} Q$ then $P \equiv_{Tr} Q$*

The clause $Tr(P) = Tr(Q)$ in the definition of completed trace equivalence is required for Theorem 1.18 to hold. To see this, consider the process P in figure 1.5 and the process Q' in Figure 1.6. They have the same set of completed traces $\{ab\}$, but they do not have the same set of traces. In fact,

$$Tr(P) = \{a, ab\} \qquad\qquad Tr(Q') = \{a, ab\} \cup \{a(c)^n |\, n \geq 1\}$$

where $(c)^n$ is the sequence consisting of n c-labelled actions.

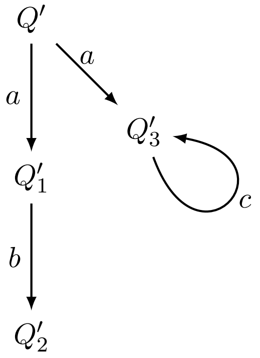

Figure 1.6: Trace equivalence and completed trace equivalence.

Definition 1.19. Let $\mathscr{A}$ be a set of atomic actions. The formulas of *Completed Trace Logic (CTL)* on $\mathscr{A}$ are defined as follows:

$$F ::= \top \;\Big|\; 0 \;\Big|\; \langle\, \mu \,\rangle F$$

We obtain the modal logic for completed trace equivalence by adding to TL-formulas the operator 0, which is true at a state if and only if it is a deadlock state:

$P \models \top$	always	
$P \models 0$	iff	$P \not\longrightarrow$
$P \models \langle\, \mu \,\rangle F$	iff	there is a P' such that $P \xrightarrow{\mu} P'$ and $P' \models F$.

$P \equiv^{CTL} Q$ if and only if for every CTL-formula F, $P \models F$ if and only if $Q \models F$.

Lemma 1.20. *Let $\mathcal{T} = \langle St, \mathscr{A}, \longrightarrow \rangle$ be an image-finite LTS. For every $n \in \mathbb{N}$, for every sequence $\mu_1, \ldots, \mu_n$ of atomic actions and for every process P,*

$$(\mu_1, \ldots, \mu_n) \in CTr(P) \textit{ if and only if } P \models \langle\, \mu_1 \,\rangle \ldots \langle\, \mu_n \,\rangle 0.$$

Proof. By induction on n.

$(n = 0)$ The empty trace ϵ belongs to $CTr(P)$ if and only if $P \not\longrightarrow$, which is equivalent to $P \models 0$.

$(n + 1)$ If $(\mu_1, \ldots, \mu_{n+1}) \in CTr(P)$ then $P \xrightarrow{\mu_1} P_1 \xrightarrow{\mu_2} \ldots \xrightarrow{\mu_{n+1}} P_{n+1}$, for some $P_1, \ldots, P_{n+1}$ such that $P_{n+1} \not\longrightarrow$. By the inductive hypothesis, $(\mu_2, \ldots, \mu_{n+1}) \in CTr(P_1)$ if and only if $P_1 \models \langle\, \mu_2 \,\rangle \ldots \langle\, \mu_{n+1} \,\rangle 0$. Then $P \models \langle\, \mu_1 \,\rangle \ldots \langle\, \mu_{n+1} \,\rangle 0$.

If $P \models \langle \mu_1 \rangle \dots \langle \mu_{n+1} \rangle 0$ then $P_1 \models \langle \mu_2 \rangle \dots \langle \mu_{n+1} \rangle 0$, for some P_1 such that $P \xrightarrow{\mu_1} P_1$. By the inductive hypothesis, $P_1 \xrightarrow{\mu_2} P_2 \xrightarrow{\mu_3} \dots \xrightarrow{\mu_{n+1}} P_{n+1}$ for some $P_3, \dots, P_{n+1}$ such that $P_{n+1} \not\rightarrow$. So, $(\mu_1, \dots, \mu_{n+1})$ is a completed trace from P.

□

Theorem 1.21. *Let $\mathcal{T} = \langle St, \mathscr{A}, \longrightarrow \rangle$ be an image-finite LTS. $P \equiv_{CTr} Q$ if and only if $P \equiv^{CTL} Q$.*

Proof. For all $n \in \mathbb{N}$ and for all traces $\mu_1, \dots, \mu_n$,

$$\begin{array}{ll} & CTr(P) = CTr(Q) \\ \text{iff} & (\mu_1, \dots, \mu_n) \in CTr(P) \text{ iff } (\mu_1, \dots, \mu_n) \in CTr(Q) \\ \text{iff} & P \models \langle \mu_1 \rangle \dots \langle \mu_n \rangle 0 \text{ iff } Q \models \langle \mu_1 \rangle \dots \langle \mu_n \rangle 0 \qquad \text{(by Lemma 1.20).} \end{array}$$

By Theorem 1.16, $Tr(P) = Tr(Q)$ if and only if $P \equiv^{TL} Q$. The result follows from the fact that a CTL-formula F is either a TL-formula or of the form $\langle \mu_1 \rangle \dots \langle \mu_n \rangle 0$, for some $n \in \mathbb{N}$ and for some $\mu_1, \dots, \mu_n$. □

We showed above that the processes P and Q in figure 1.5 are not completed trace equivalent. In fact, the formula $\langle a \rangle 0$ is true at Q and false at P.
Now consider the processes R and T in figure 1.7.

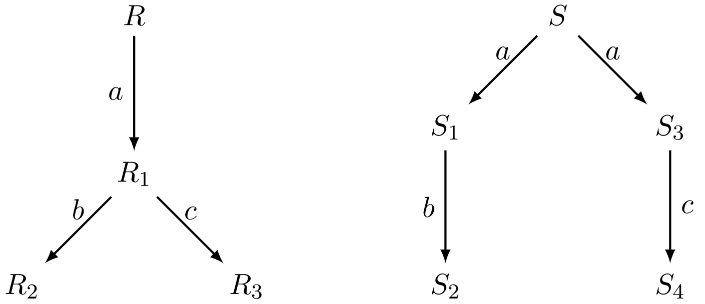

Figure 1.7: Completed trace equivalent processes.

The processes R and S have the same set of traces $\{a, ab, ac\}$ and the same set of completed traces $\{ab, ac\}$. However, if P performs an a-labelled transition then it is still possible to choose whether to do b or c. In contrast, the process Q always reaches a state where either b or c is not available.

1.3.3 Simulation equivalence

Unlike trace-based equivalences, simulation equivalence can observe the branching time of a process. A simulation relation is obtained by breaking the symmetry of bisimulation; in particular, we drop the second clause in the definition of bisimulation.[25]

Definition 1.22. Let $\mathcal{T} = \langle St, \mathscr{A}, \longrightarrow \rangle$ be an *LTS*. A relation $\mathcal{R} \subseteq St \times St$ is a *simulation* if $P \, \mathcal{R} \, Q$ implies:

- for every $\mu \in \mathscr{A}$, $P' \in St$, if $P \xrightarrow{\mu} P'$ then $Q \xrightarrow{\mu} Q'$ and $P' \, \mathcal{R} \, Q'$, for some $Q' \in St$.

We say that P is *simulated* by Q ($P \lesssim Q$) if there is a simulation $\mathcal{R}$ such that $P \, \mathcal{R} \, Q$. P and Q are *simulation equivalent* ($P \eqsim Q$) if $P \lesssim Q$ and $Q \lesssim P$.

Theorem 1.23. *Let $\mathcal{T} = \langle St, \mathscr{A}, \longrightarrow \rangle$ be an LTS.*

1. *$\lesssim$ is a preorder.*
2. *$\eqsim$ is an equivalence relation.*
3. *$\lesssim$ is the largest simulation relation on St.*

Proof.

1. We prove that $\lesssim$ is reflexive and transitive.
   - Let $\mathcal{I}$ be the identity relation on the processes in $\mathcal{T}$. If $P \xrightarrow{\mu} P'$ then $P \xrightarrow{\mu} P'$ and $(P', P') \in \mathcal{I}$. Therefore, $\mathcal{I}$ is a simulation, which implies that $P \lesssim P$.
   - If $P \lesssim P_1 \lesssim P_2$ then there are two simulation relations $\mathcal{R}_1$ and $\mathcal{R}_2$ such that $P \mathcal{R}_1 P_1$ and $P_1 \mathcal{R}_2 P_2$. In order to prove that $P \lesssim P_2$ we show that $\mathcal{R}_3 = \mathcal{R}_1 \circ \mathcal{R}_2 = \{(P, P_2) |$ there exists a P_1 such that $P \mathcal{R}_1 P_1$ and $P_1 \mathcal{R}_2 P_2\}$ is a simulation. Let $P \mathcal{R}_3 P_2$ and let $P \xrightarrow{\mu} P'$. Then $P_1 \xrightarrow{\mu} P_1'$ and $P' \mathcal{R}_1 P_1'$, for some P_1 such that $P \mathcal{R}_1 P_1 \mathcal{R}_2 P_2$ and for some P_1'. It follows from $P_1 \mathcal{R}_2 P_2$ that there is a P_2' such that $P_2 \xrightarrow{\mu} P_2'$ and $P_1' \mathcal{R}_2 P_2'$. Therefore, $P' \mathcal{R}_3 P_2'$ and the relation $\mathcal{R}_3$ is a simulation.
2. It follows from the previous point that $\eqsim$ is reflexive and transitive. Symmetry is a direct consequence of the definition of $\eqsim$.
3. Suppose $P \lesssim Q$. Then there is a simulation $\mathcal{R}$ such that $P \, \mathcal{R} \, Q$. If $P \xrightarrow{\mu} P'$ then $Q \xrightarrow{\mu} Q'$ for some Q' such that $P' \, \mathcal{R} \, Q'$. Therefore, $P' \lesssim Q'$ and we can conclude that $\lesssim$ is itself a simulation relation.
   If $\mathcal{R}$ is a simulation on St then by the definition of $\lesssim$ we have that $\mathcal{R} \subseteq \lesssim$. So, the simulation preorder is also the largest simulation relation on St.

□

[25] The notion of simulation in theoretical computer was developed earlier than the notion of bisimulation, see (Milner 1971) and (Park 1981).

The processes R and S in Figure 1.7 are completed trace equivalent, but they are not simulation equivalent.
The relation $\{(S,R),(S_1,R_1),(S_3,R_1),(S_2,R_2),(S_4,R_3)\}$ is a simulation relation, thus the process S is simulated by R. Now suppose that there exists a simulation $\mathcal{R}$ such that $R\,\mathcal{R}\,S$. It follows from $R \xrightarrow{a} R_1$ that either $R_1\,\mathcal{R}\,S_1$ or $R_1\,\mathcal{R}\,S_3$, being S_1 and S_3 the only states reachable by S through an a-labelled transition. In the first case we have that $R_1 \xrightarrow{c} R_3$ but $S_1 \not\xrightarrow{c}$; in the second one, S_3 cannot match the b-labelled transition from R_1. This contradicts the assumption that $R \simeq S$.

Lemma 1.24. *Let $\mathcal{T} = \langle St, \mathscr{A}, \longrightarrow\rangle$ be an LTS. If $P \lesssim Q$ and $P \xrightarrow{\mu_1} P_1 \xrightarrow{\mu_2} \dots \xrightarrow{\mu_n} P_n$ then there are $Q_1 \dots Q_n$ such that $Q \xrightarrow{\mu_1} Q_1 \xrightarrow{\mu_2} \dots \xrightarrow{\mu_n} Q_n$ and $P_n \lesssim Q_n$.*

Proof. By induction on $n \in \mathbb{N}$. The base case holds by definition. As for the inductive step, suppose that $P \lesssim Q$ and $P \xrightarrow{\mu_1} \dots \xrightarrow{\mu_{n+1}} P_{n+1}$. By the inductive hypothesis, there are $Q_1 \dots Q_n$ such that $Q \xrightarrow{\mu_1} \dots \xrightarrow{\mu_n} Q_n$ and $P_n \lesssim Q_n$. By Theorem 1.23 (point 3), $\lesssim$ is a simulation itself, so it follows from $P_n \xrightarrow{\mu_{n+1}} P_{n+1}$ that $Q_n \xrightarrow{\mu_{n+1}} Q_{n+1}$ and $P_{n+1} \lesssim Q_{n+1}$. Hence, $Q \xrightarrow{\mu_1} \dots \xrightarrow{\mu_n} Q_n \xrightarrow{\mu_{n+1}} Q_{n+1}$ and $P_{n+1} \lesssim Q_{n+1}$.

□

Theorem 1.25. *Let $\mathcal{T} = \langle St, \mathscr{A}, \longrightarrow\rangle$ be an LTS. If $P \eqsim Q$ then $P \equiv_{Tr} Q$.*

Proof. As a consequence of Lemma 1.24, $P \lesssim Q$ implies $Tr(P) \subseteq Tr(Q)$. The result follows from the symmetry of $\eqsim$.

□

By Theorem 1.25, simulation equivalence is included in trace equivalence. The inclusion is strict: the processes in Figure 1.7 are discriminated under simulation equivalence, but they are completed trace equivalent and thus trace equivalent.

Definition 1.26. Let $\mathscr{A}$ be a set of atomic actions. The formulas of *Simulation Logic (SL)* on $\mathscr{A}$ are defined as follows:

$$F ::= \top \;\Big|\; F_1 \wedge F_2 \;\Big|\; \langle\,\mu\,\rangle F$$

Thus, SL is obtained by dropping negation from Hennessy-Milner logic. The semantic of SL-formulas is inductively defined as follows:

$P \models \top$	always	
$P \models F_1 \wedge F_2$	iff	$P \models F_1$ and $P \models F_2$
$P \models \langle\,\mu\,\rangle F$	iff	there is a P' such that $P \xrightarrow{\mu} P'$ and $P' \models F$.

$P \sqsubseteq^{SL} Q$ if and only if for every SL-formula F, $P \models F$ implies $Q \models F$.
$P \equiv^{SL} Q$ if $P \sqsubseteq^{SL} Q$ and $Q \sqsubseteq^{SL} P$.

Theorem 1.27. *Let $\mathcal{T} = \langle St, \mathscr{A}, \longrightarrow \rangle$ be an image-finite LTS. $P \lesssim Q$ if and only if $P \sqsubseteq^{SL} Q$.*

Proof.

($\Rightarrow$) We prove by structural induction on F that $P \lesssim Q$ implies that for every SL-formula F, if $P \models F$ then $Q \models F$. Let $P \lesssim Q$.

($F = \top$) It always holds that $P \models \top$ and $Q \models \top$.

($F = F_1 \wedge F_2$) $P \models F_1 \wedge F_2$ implies $P \models F_1$ and $P \models F_2$. By the inductive hypothesis, $Q \models F_1$ and $Q \models F_2$. As a consequence, $Q \models F_1 \wedge F_2$.

($F = \langle \mu \rangle F'$) If $P \models \langle \mu \rangle F'$ then there exists a P' such that $P \xrightarrow{\mu} P'$ and $P' \models F$. It follows from $P \lesssim Q$ and from Theorem 1.23 that $Q \xrightarrow{\mu} Q'$ for some Q' such that $P' \lesssim Q'$. By the inductive hypothesis we have that $Q' \models F'$, which in turn implies that $Q \models \langle \mu \rangle F'$.

($\Leftarrow$) We prove that $\sqsubseteq^{SL}$ is a simulation.
Let $P \sqsubseteq^{SL} Q$ and $P \xrightarrow{\mu} P'$. Then $\langle \mu \rangle \top$ is true at P, which implies that $Q \models \langle \mu \rangle \top$. Hence, the set $\{Q_1, \ldots, Q_n\}$ of Q's μ-derivatives is not empty. By the image-finiteness hypothesis, this set is finite. Suppose that $P' \not\sqsubseteq^{SL} Q_i$ for all $i \in \{1, \ldots, n\}$. Then there exists a formula F_i such that $P' \models F_i$ and $Q_i \not\models F_i$ for every $i \in \{1, \ldots, n\}$. Therefore, $P \models \langle \mu \rangle (F_1 \wedge \ldots \wedge F_n)$, which implies that $Q \models \langle \mu \rangle (F_1 \wedge \ldots \wedge F_n)$, by the hypothesis that $P \sqsubseteq^{SL} Q$. However, the only states reachable by Q through a μ-labelled transition are in $\{Q_1, \ldots, Q_n\}$ and for every i we have that $Q_i \not\models F_i$. Thus, $Q' \not\models (F_1 \wedge \ldots \wedge F_n)$ for every Q' such that $Q \xrightarrow{\mu} Q'$, in contradiction with the assumption that $Q \models \langle \mu \rangle (F_1 \wedge \ldots \wedge F_n)$.

□

As a corollary of Theorem 1.27 we have that:

Corollary 1.28. *Let $\mathcal{T} = \langle St, \mathscr{A}, \longrightarrow \rangle$ be an image-finite LTS. $P \eqsim Q$ if and only if $P \equiv^{SL} Q$.*

We proved above with an argument by contradiction that the process R in Figure 1.7 cannot be simulated by S. By Corollary 1.28 it suffices to exhibit the SL-formula $\langle a \rangle (\langle b \rangle \top \wedge \langle b \rangle \top)$, which is true at R but false at S.

Just like trace equivalence, simulation equivalence is insensitive to deadlock. Let us consider again the processes P and R in Figure 1.5. The relation:

$$\{(P, Q), (P_1, Q_1), (P_2, Q_2)\}$$

is a simulation, so $P \simeq Q$. The other direction holds as well:

$$\{(Q, P), (Q_1, P_1), (Q_2, P_2), (Q_3, P_1)\}$$

is a simulation. We conclude that P and Q are simulation equivalent, even if the first process never terminates after performing an a-labelled transition, while the second one does.

1.3.4 Completed simulation equivalence

In Section 1.3.2 we presented completed trace equivalence as a deadlock-sensitive refinement of trace equivalence. Completed simulation equivalence plays the same role with respect to simulation equivalence.

Definition 1.29. Let $\mathcal{T} = \langle St, \mathscr{A}, \longrightarrow\rangle$ be an *LTS*. A simulation $\mathcal{R} \subseteq St \times St$ is a *completed simulation* if $P \mathcal{R} Q$ implies:

- if $P \not\longrightarrow$ then $Q \not\longrightarrow$.

We say that P is *completed simulated* by Q ($P \lesssim^c Q$) if there is a complete simulation $\mathcal{R}$ such that $P \mathcal{R} Q$. P and Q are *completed simulation equivalent* ($P \eqsim^c Q$) if $P \lesssim^c Q$ and $Q \lesssim^c P$.

Theorem 1.30. *Let $\mathcal{T} = \langle St, \mathscr{A}, \longrightarrow\rangle$ be an LTS. If $P \eqsim^c Q$ then $P \eqsim Q$.*

Proof. If $P \eqsim^c Q$ then there are two completed simulations $\mathcal{R}_1$ and $\mathcal{R}_2$ such that $P \mathcal{R}_1 Q$ and $Q \mathcal{R}_2 P$. By definition, $\mathcal{R}_1$ and $\mathcal{R}_2$ are simulations, so $P \lesssim Q$ and $Q \lesssim P$ and the result follows. □

The opposite direction of Theorem 1.30 ($P \eqsim Q$ implies $P \eqsim^c Q$) does not hold.
The processes P and Q in Figure 1.5 are simulation equivalent and it also holds that $P \lesssim^c Q$, for $\{(P, Q), (P_1, Q_1), (P_2, Q_2)\}$ is a completed simulation. Suppose that there exists a completed simulation relation $\mathcal{R}$ such that $Q \mathcal{R} P$. It follows from $Q \xrightarrow{a} Q_3$ that P_1 must be in the relation $\mathcal{R}$ with Q_3, being P_1 the only state that P can reach via an a-labelled transition. Yet, the process Q_2 is stopped while P_1 can move. This contradicts the assumption that $Q_3 \mathcal{R} P_1$, so Q cannot be completed simulated by P. Therefore, P and Q are discriminated under completed simulation equivalence.

Theorem 1.31. *Let $\mathcal{T} = \langle St, \mathscr{A}, \longrightarrow\rangle$ be an LTS. If $P \eqsim^c Q$ then $P \equiv_{CTr} Q$.*

Proof. By Theorem 1.30, if $P \eqsim^c Q$ then $P \eqsim Q$, which implies by Theorem 1.25 that P and Q are trace equivalent.
Analogously to Lemma 1.24, it is easy to check that $P \lesssim^c Q$ and $P \xrightarrow{\mu_1} P_1 \xrightarrow{\mu_2} \dots \xrightarrow{\mu_n} P_n$ implies that there are processes $Q_1 \dots Q_n$ such that $Q \xrightarrow{\mu_1} Q_1 \xrightarrow{\mu_2} \dots \xrightarrow{\mu_n} Q_n$ and $P_n \lesssim^c Q_n$. Therefore, if $P \xrightarrow{\mu_1} P_1 \xrightarrow{\mu_2} \dots \xrightarrow{\mu_n} P_n \not\longrightarrow$ then there are $Q_1 \dots Q_n$ such that $Q \xrightarrow{\mu_1} Q_1 \xrightarrow{\mu_2} \dots \xrightarrow{\mu_n} Q_n$ and $P_n \lesssim^c Q_n$, which implies that $Q_n \not\longrightarrow$. So, $P \lesssim^c Q$ implies $CTr(P) \subseteq CTr(Q)$ and the result follows by the symmetry of $\eqsim^c$. □

We proved that the processes in Figure 1.7 are not simulation equivalent, so by Theorem 1.30 they are not completed simulation equivalent. However, R and S are completed trace equivalent, thus by Theorem 1.31 we have that completed simulation equivalence is strictly finer than completed trace equivalence.

Definition 1.32. Let $\mathscr{A}$ be a set of atomic actions. The formulas of *Completed Simulation Logic (CSL)* on $\mathscr{A}$ are defined as follows:

$$F ::= \top \;\Big|\; 0 \;\Big|\; F_1 \wedge F_2 \;\Big|\; \langle\, \mu \,\rangle F$$

CSL-formulas are obtained by adding the formula 0 of Completed Trace Logic to the formulas of Simulation Logic. The truth conditions of these formulas remain unchanged:

$P \models \top$ always

$P \models 0$ iff $P \not\longrightarrow$

$P \models F_1 \wedge F_2$ iff $P \models F_1$ and $P \models F_2$

$P \models \langle\, \mu \,\rangle F$ iff there is a P' such that $P \xrightarrow{\mu} P'$ and $P' \models P$.

$P \sqsubseteq^{CSL} Q$ if and only if for every CSL-formula F, $P \models F$ implies $Q \models F$.
$P \equiv^{CSL} Q$ if $P \sqsubseteq^{CSL} Q$ and $Q \sqsubseteq^{CSL} P$.

Lemma 1.33. *Let $\mathcal{T} = \langle St, \mathscr{A}, \longrightarrow \rangle$ be an image-finite LTS. $P \lesssim^c Q$ if and only if $P \sqsubseteq^{CSL} Q$.*

Proof.

(⇒) We prove by structural induction on F that $P \lesssim^c Q$ implies that for every CSL-formula F, if $P \models F$ then $Q \models F$. The proof is analogous to the one of Theorem 1.27 as far as the subset of SL-formulas is concerned, so we only consider the case when $F = 0$.
Suppose that $P \lesssim^c Q$ and let $P \models 0$. Then $P \not\longrightarrow$ and it follows by the definition of completed simulation that $Q \not\longrightarrow$. Therefore, $Q \models 0$.

($\Leftarrow$) We prove that $\sqsubseteq^{CSL}$ is a completed simulation.

By using the fact that SL-formulas are a subset of CSL-formulas we can prove that $\sqsubseteq^{CSL}$ is a simulation (Theorem 1.27, right to left direction). Suppose that $P \sqsubseteq^{CSL} Q$. If $P \not\longrightarrow$ then $P \models 0$, which implies that $Q \models 0$. Thus, $Q \not\longrightarrow$ and $\sqsubseteq^{CSL}$ satisfies the specific clause of the definition of completed simulation as well.

□

It follows from Lemma 1.33 hat completed simulation equivalence coincides with the equivalence on CSL-formulas on image-finite LTSs:

Theorem 1.34. *Let $\mathcal{T} = \langle St, \mathscr{A}, \longrightarrow \rangle$ be an image-finite LTS. $P \approx^c Q$ if and only if $P \equiv^{CSL} Q$.*

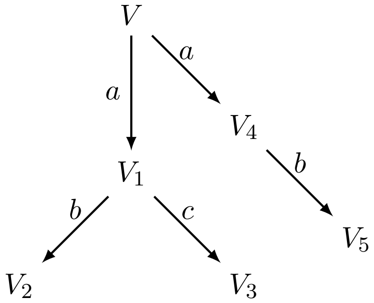

Figure 1.8: A process completed simulation equivalent to R (Fig. 1.7).

Consider the process V in Figure 1.8 and the process R in Figure 1.7. Both the relation:

$$\mathcal{R}_1 = \{(R, V), (R_1, V_1), (R_2, V_2), (R_3, V_3)\}$$

and the relation:

$$\mathcal{R}_2 = \{(V, R), (V_1, R_1), (V_2, R_2), (V_3, R_3), (V_4, R_1), (V_5, R_2)\}$$

are completed simulations. Therefore, R and V are completed simulation equivalent, even if we can observe a difference in their behavior: every time R performs an a-labelled transition, both the action b and the action c can be done; if V moves to V_4 by doing a, then c is not available anymore.

1.3.5 Ready simulation equivalence

Completed simulation refines the definition of simulation by allowing us to observe whether related states are both stopped. Ready simulation further improves the sensitivity of simulation relations: related processes are required to have the same set of available atomic actions they can perform.

Ready simulation equivalence was introduced in (Larsen and Skou 1991) under the name of $\frac{2}{3}$- bisimilarity; later, it was studied in (Bloom, Istrail, and Meyer 1995) under the name of *GSOS trace congruence*.

Definition 1.35. Let $\mathcal{T} = \langle St, \mathscr{A}, \longrightarrow \rangle$ be an *LTS*. A simulation $\mathcal{R} \subseteq St \times St$ is a *ready simulation* if $P \mathcal{R} Q$ implies:

- for every $\mu \in \mathscr{A}$, if $P \overset{\mu}{\not\longrightarrow}$ then $Q \overset{\mu}{\not\longrightarrow}$.

We say that P is *ready simulated* by Q ($P \lesssim^r Q$) if there is a ready simulation $\mathcal{R}$ such that $P \mathcal{R} Q$. P and Q are *ready simulation equivalent* ($P \eqsim^r Q$) if $P \lesssim^r Q$ and $Q \lesssim^r P$.

The process V in Figure 1.8 is not ready simulated by the process R in Figure 1.7. Suppose that this is false, that is, suppose that there is a ready simulation relation $\mathcal{R}$ such that $V \mathcal{R} R$. The process V reaches V_4 by performing an a-labelled transition and R_1 is the only state such that $R \xrightarrow{a} R_1$, so V_4 must be in the relation $\mathcal{R}$ with R_1. However, the process V_4 cannot perform a c-labelled transition, while $R_1 \xrightarrow{c}$. This contradicts the assumption that there exists a ready simulation $\mathcal{R}$ such that $V \mathcal{R} R$.

Conversely, ready simulation equivalence is included in completed simulation equivalence.

Theorem 1.36. *Let $\mathcal{T} = \langle St, \mathscr{A}, \longrightarrow \rangle$ be an LTS. If $P \eqsim^r Q$ then $P \eqsim^c Q$.*

Proof. If $P \lesssim^r Q$ then there is a ready simulation $\mathcal{R}$ such that $P \mathcal{R} Q$. If $P \not\longrightarrow$ then $P \overset{\mu}{\not\longrightarrow}$ for every atomic action μ. By the definition of ready simulation, $Q \overset{\mu}{\not\longrightarrow}$ for every μ, that is, $Q \not\longrightarrow$. Thus, $\mathcal{R}$ is also a completed simulation. □

Definition 1.37. Let $\mathscr{A}$ be a set of atomic actions. The formulas of *Ready Simulation Logic (RSL)* on $\mathscr{A}$ are defined as follows:

$$F ::= \top \;\Big|\; \neg\mu \;\Big|\; F_1 \wedge F_2 \;\Big|\; \langle\, \mu \,\rangle F$$

Ready Simulation Logic adds the negation operator to the ones of SL. However, only a restricted use of $\neg$ is allowed, i.e. it is only possible to negate the possibility of performing an atomic action. In fact, the formula $\neg\mu$ is semantically equivalent to the HML-formula $\neg\langle\, \mu \,\rangle\top$.

$P \models \top$	always	
$P \models \neg\mu$	iff	$P \overset{\mu}{\not\longrightarrow}$
$P \models F_1 \wedge F_2$	iff	$P \models F_1$ and $P \models F_2$
$P \models \langle\, \mu \,\rangle F$	iff	there is a P' such that $P \xrightarrow{\mu} P'$ and $P' \models F$.

$P \sqsubseteq^{RSL} Q$ if and only if for every RSL-formula F, $P \models F$ implies $Q \models F$.
$P \equiv^{RSL} Q$ if $P \sqsubseteq^{RSL} Q$ and $Q \sqsubseteq^{RSL} P$.

Theorem 1.38. *Let $\mathcal{T} = \langle St, \mathscr{A}, \longrightarrow \rangle$ be an image-finite LTS. $P \lesssim^r Q$ if and only if $P \sqsubseteq^{RSL} Q$.*

Proof.

($\Rightarrow$) The proof is by structural induction on F and it is analogous to the one of Theorem 1.27, as far as the formulas of Simulation Logic are concerned. Now consider the case when $F = \neg\mu$. If $P \lesssim^c Q$ and $P \models \neg\mu$ then $P \overset{\mu}{\not\longrightarrow}$. By the definition of ready simulation we have that $Q \overset{\mu}{\not\longrightarrow}$, which implies that $Q \models \neg\mu$.

($\Leftarrow$) We prove that $\sqsubseteq^{CSL}$ is a ready simulation.
$\sqsubseteq^{RSL}$ is a simulation, being the SL-formulas included in RSL-formulas (see Theorem 1.27, right-to-left direction). If $P \sqsubseteq^{RSL} Q$ and $P \overset{\mu}{\not\longrightarrow}$ then $P \models \neg\mu$. Therefore, $Q \models \neg\mu$, which is equivalent to $Q \not\longrightarrow$. So we can conclude that $\sqsubseteq^{RSL}$ is a ready simulation.

□

Let us consider again the processes R and V in Figure 1.7 and in Figure 1.8, respectively. We proved that $V \not\lesssim^r R$. In fact, the RSL-formula $\langle\, a \,\rangle\neg c$ is true at V and false at R.
As a consequence of Theorem 1.38, ready simulation equivalence and the equivalence on RSL-formulas coincide.

Corollary 1.39. *Let $\mathcal{T} = \langle St, \mathscr{A}, \longrightarrow \rangle$ be an image-finite LTS. $P \eqsim^r Q$ if and only if $P \equiv^{RSL} Q$.*

Ready simulation equivalence fails to distinguish the two processes in Figure 1.9. The relations:

$$\mathcal{R}_1 = \{(T, U), (T_1, U_1), (T_2, U_2), (T_3, U_3), (T_4, U_4), (T_5, U_5)\}$$
$$\mathcal{R}_2 = \{(U, T), (U_1, T_1), (U_2, T_2), (U_3, T_3), (U_4, T_4), (U_5, T_5), (U_6, T_1), (U_7, T_2), (U_8, T_3)\}$$

are ready simulations, hence $T \lesssim^r U$ and $U \lesssim^r T$.

1.3.6 Bisimilarity and isomorphism

We can look at bisimilarity as the last refinement of simulation equivalence. It is easy to check that a relation $\mathcal{R}$ is a bisimulation if and only if both $\mathcal{R}$ and $\mathcal{R}^{-1} = \{(P, Q) | \, Q \, \mathcal{R} \, P\}$ are simulations. This requirement allows bisimilarity to distinguish the two processes in Figure 1.9. The process T satisfies the HML-formula $[\, a \,]\langle\, b \,\rangle\langle\, d \,\rangle\top$, while U does not; by

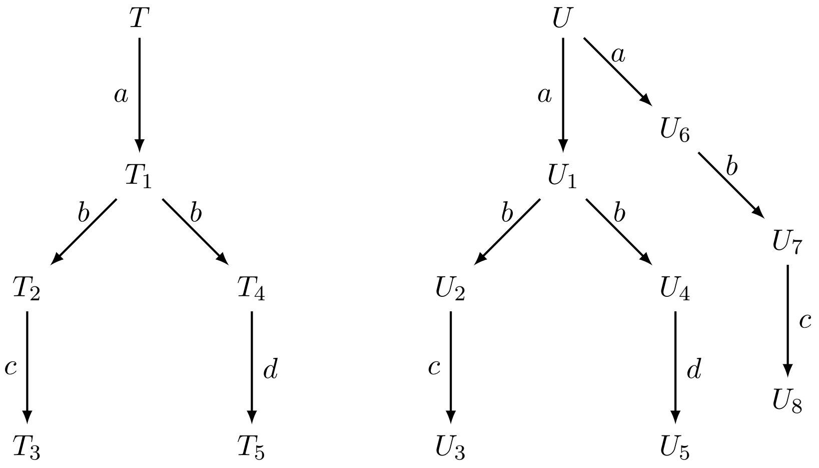

Figure 1.9: Ready simulation equivalent processes.

Theorem 1.12, T and U are not bisimilar.
Conversely, bisimilarity implies ready simulation equivalence.

Theorem 1.40. *Let $\mathcal{T} = \langle St, \mathscr{A}, \longrightarrow\rangle$ be an LTS. If $P \sim Q$ then $P \simeq^r Q$.*

Proof. If $P \sim Q$ then there is a bisimulation $\mathcal{R}$ such that $P\,\mathcal{R}\,Q$. Both $\mathcal{R}$ and $\mathcal{R}^{-1}$ are simulation relations, therefore $P' \xrightarrow{\mu}$ if and only if $Q' \xrightarrow{\mu}$, for every P', Q' such that $P'\,\mathcal{R}\,Q'$. As a consequence, both $\mathcal{R}$ and $\mathcal{R}^{-1}$ are ready simulation relations. □

To show that bisimilarity is not an over-discriminating equivalence relation on processes, we prove that isomorphism on graphs is strictly included in bisimilarity.

Definition 1.41. Let $\mathcal{T} = \langle St, \mathscr{A}, \longrightarrow\rangle$ be an *LTS* and let P and Q be processes in $\mathcal{T}$. A function $f : St_P \to St_Q$ is an *isomorphism* between P and Q if:

- f is a bijection,
- $f(P) = Q$,
- $P' \xrightarrow{\mu} P''$ if and only if $f(P') \xrightarrow{\mu} f(P'')$, for every μ and for all processes P', P'' reachable from P.

We say that P and Q are *isomorphic* ($P \cong Q$) if there is an isomorphism f between P and Q.

Theorem 1.42. *Let $\mathcal{T} = \langle St, \mathscr{A}, \longrightarrow\rangle$ be an LTS. If $P \cong Q$ then $P \sim Q$.*

Proof. We prove that $\cong$ is a bisimulation.
Suppose that $P \cong Q$. Then there is an isomorphism f such that $f(P) = Q$. If $P \xrightarrow{\mu} P'$

then there is a Q' such that $Q = f(P) \xrightarrow{\mu} f(P') = Q'$. Let $f' : St_{P'} \to St_{Q'}$ be the restriction of f to the LTS reachable from P'. The function f' is an isomorphism between P' and Q':

- the graph of f' is included into the graph of f, so the bijectivity of f implies the bijectivity of f',
- $f'(P') = Q'$,
- for every μ and for all processes P'', P''' reachable from P', $P'' \xrightarrow{\mu} P'''$ if and only if $f(P'') \xrightarrow{\mu} f(P''')$ if and only if $f'(P'') \xrightarrow{\mu} f'(P''')$.

Hence, $P' \cong Q'$.
The function f is a bijection, so $Q \xrightarrow{\mu} Q'$ implies that $Q' = f(P')$ for some $P' \in St_P$. Then $Q \xrightarrow{\mu} Q'$ if and only if $f(P) \xrightarrow{\mu} f(P')$ if and only if $P \xrightarrow{\mu} P'$. As above, the restriction of f to the LTS reachable from P' is an isomorphism, so $P' \cong Q'$ and the second condition in the definition of bisimulation holds as well. □

We proved that the processes X and Y in Figure 1.3 are bisimilar: although some differences between their structures, they have the same behavior. However, X and Y are not isomorphic: the LTS reachable from the first process has less states than the LTS reachable from the second one, so there cannot be any bijection between X and Y.

1.3.7 The spectrum of equivalences

Figure 1.10 summarizes the results presented in this chapter. An arrow from a relation to another one means that the former is included in the latter; the arrows are labelled with the number of the theorem proving the inclusion. All these implications between equivalence relations on processes are strict, as the counterexamples provided in the previous sections showed.

1.4 A Calculus of Communicating Systems

The process calculus CCS (*Calculus of Communicating Systems*) is a language whose terms represent the behavior of concurrent programs. CCS was first introduced in (Milner 1980) and its theory was further developed in (Milner 1989).[26]
One and only one LTS is associated to every term of CCS by means of an operational semantic (Plotkin 2004b, Plotkin 2004a), a syntax-driven semantic defined by the rules of the calculus. Every operator of CCS has its own (possibly empty) set of axioms or inference rules, which allow us to derive the transitions from a term to its subterms.

[26] We present here a slightly modified version of the original CCS, following the lines of (Sangiorgi 2012a).

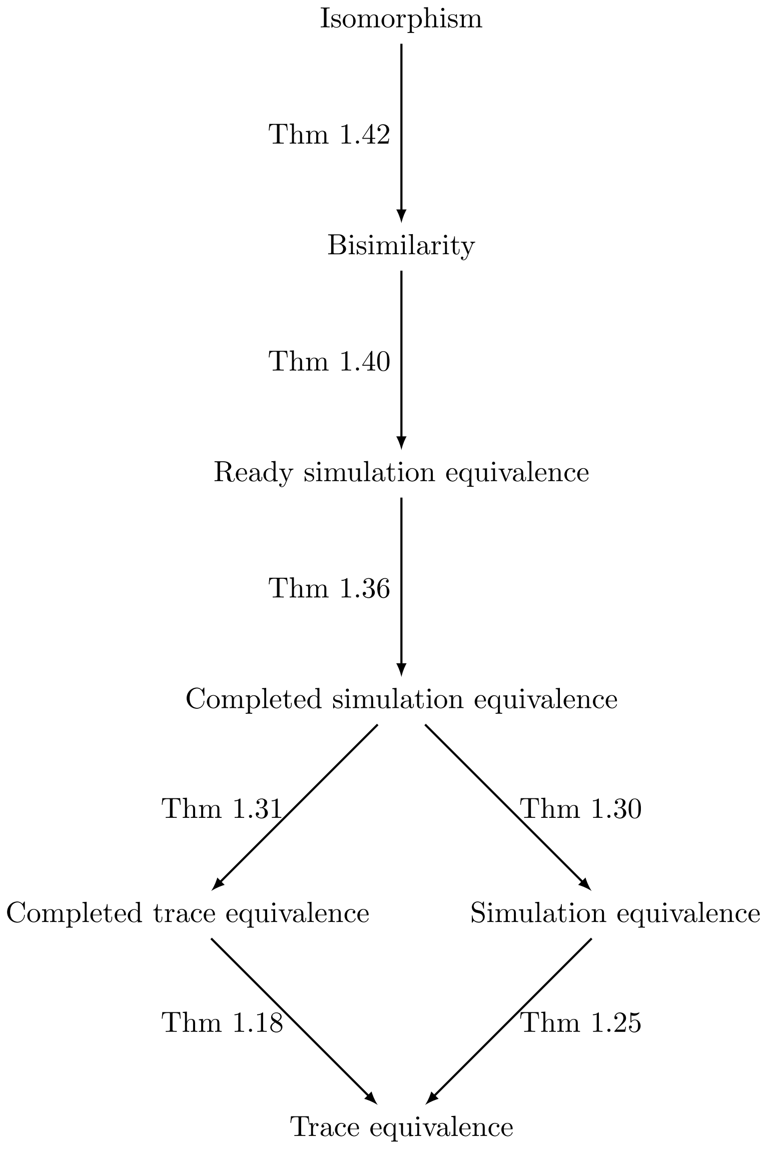

Figure 1.10: The spectrum of equivalences.

Hence, the semantic of CCS gives rise to an huge LTS, whose set of states is exactly the set of CCS-terms.

Let $\mathscr{N}$ be a set of atomic actions.

- $\overline{\mathscr{N}}$ is the set of its complementary names (or *conames*), i.e. the set $\{\bar{a} |\, a \in \mathscr{N}\}$ where $a = \bar{\bar{a}}$,
- $\mathscr{A} = \mathscr{N} \cup \overline{\mathscr{N}}$.

The set $\mathscr{N}$ is interpreted as the set of actions that programs may take as input, while $\overline{\mathscr{N}}$

is the set of their complementary output actions. We let $a, b, c \ldots$ denote input actions and $\bar{a}, \bar{b}, \bar{c}, \ldots$ denote output actions, where $a = \bar{\bar{a}}$. We use $\alpha, \beta, \ldots$ to range over $\mathscr{A}$, where $\alpha = \bar{\bar{\alpha}}$.

CCS features operators that allows us to describe the evolution of a nondeterministic system whose possible inputs and outputs are in $\mathscr{A}$.

The terms of CCS on $\mathscr{N}$ are defined by the following grammar:

$$P ::= \mathbf{0} \;\Big|\; \alpha.P \;\Big|\; P_1 + P_2 \;\Big|\; P_1 \,|\, P_2 \;\Big|\; (\nu a)P \;\Big|\; A$$

where A is a constant (we suppose to have at our disposal infinitely many names for constants).

$$\frac{}{\alpha.P \xrightarrow{\alpha} P}\ (\text{pref})$$

$$\frac{P_1 \xrightarrow{\mu} P_1'}{P_1 + P_2 \xrightarrow{\mu} P_1'}\ (\text{sumL}) \qquad \frac{P_2 \xrightarrow{\mu} P_2'}{P_1 + P_2 \xrightarrow{\mu} P_2'}\ (\text{sumR})$$

$$\frac{P_1 \xrightarrow{\mu} P_1'}{P_1 \,|\, P_2 \xrightarrow{\mu} P_1' \,|\, P_2}\ (\text{parL}) \qquad \frac{P_2 \xrightarrow{\mu} P_2'}{P_1 \,|\, P_2 \xrightarrow{\mu} P_1 \,|\, P_2'}\ (\text{parR})$$

$$\frac{P_1 \xrightarrow{\alpha} P_1' \qquad P_2 \xrightarrow{\bar{\alpha}} P_2'}{P_1 \,|\, P_2 \xrightarrow{\tau} P_1' \,|\, P_2'}\ (\text{sync})$$

$$\frac{P \xrightarrow{\mu} P'}{(\nu a)P \xrightarrow{\mu} P'}\ (\text{res})_{a \neq \mu \neq \bar{a}} \qquad \frac{P \xrightarrow{\mu} P'}{A \xrightarrow{\mu} P'}\ (\text{cons})_{A \overset{def}{=} P}$$

Figure 1.11: SOS for processes in CCS.

The operational semantic of CCS on $\mathscr{N}$ is defined by the LTS $\langle \text{CCS-terms}, \mathscr{A} \cup \{\tau\}, \longrightarrow \rangle$, where the transition relation $\longrightarrow$ is given by the rules in Figure 1.11 and τ is a fresh name of action. Let us briefly describe the operators of CCS.

The first operator is the *nil* operator $\mathbf{0}$, representing a terminated process. The process $\mathbf{0}$ cannot move, which is the reason why there are no rules for it in the operational semantic of CCS.

The *prefix* operator allows us to capture labelled state transitions. Given an input or output atomic action α and a process P, the process $\alpha.P$ is the process that sequentially performs a transition labelled with the prefix α and becomes the process P. The rule for prefixing is actually an axiom, named (pref).

The third operator of CCS is the *sum* (or *nondeterministic choice*) operator $+$. The process $P_1 + P_2$ is the process which nondeterministically behaves either as P_1 or as P_2. The rules (sumL) and (sumR) describe the first and the second case, respectively. The prefix and sum operators are so called *dynamic operators*, for they disappear in the right-hand side of the relative rules' conclusion. Conversely, the *parallel* operator $\mid$ and the *restriction* operator (νa) are not "lost after use", so they are said to be *static operators*.

The process $P_1 \mid P_2$ is the parallel composition of the processes P_1 and P_2. When two processes are executed in parallel we can have two different cases:

- the process P_1 (respectively: P_2) performs a transition labelled with an input or output atomic action and $P_1 \mid P_2$ evolves into a process which is the parallel composition of the derivative of P_1 (respectively: the process P_1) and the process P_2 (respectively: the derivative of P_2);

- the two processes perform complementary input-output actions which synchronize. In this case, $P_1 \mid P_2$ evolves into a process which is the parallel composition of P_1's continuation and P_2's continuation. The synchronization give rise to an internal state transition, labelled with τ.

The first case represents P_1's (respectively: P_2's) possibility of performing its own transitions, so to communicate with external processes, without affecting the parallel structure of the system. The second one represent the two components' interaction with each other, provided that they are capable of performing complementary actions.

Given a process P and an atomic action $a \in \mathscr{N}$, the restriction $(\nu a)P$ stops the execution of a state transition labelled with a or $\bar{a}$. As a consequence, if $P = P_1 \mid P_2$ then the restriction on a forces the synchronization between P_1 and P_2, in case one of them requires the input a and the other one offers the complementary output $\bar{a}$. Restriction on an atomic action aims at representing private communication between the restricted processes: acting as a binder on a and $\bar{a}$, these input-output actions only are available for synchronization, since they only are observable to processes under the scope of (νa).

The last construct of CCS is the *constant*. We assume that there is a countably infinite set of constant names, ranged over by $A, B, C, \ldots$ and that every constant is defined by an equation of the form $A = P$, where P is a CCS process.[27] Constants allow us to define CCS-terms with an infinitary behavior (such as processes that can reach infinitely many states and processes with loops, as illustrated below), by admitting recursive definitions of constants.

For notational convenience, we adopt the following order of precedence among the operators of CCS: restriction > prefix > parallel composition > sum.

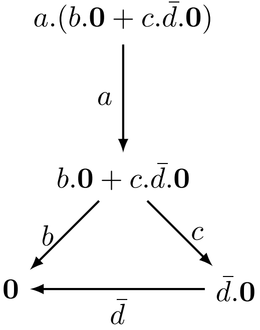
$a.(b.\mathbf{0} + c.\bar{d}.\mathbf{0})$
a
$b.\mathbf{0} + c.\bar{d}.\mathbf{0}$
b
c
$\mathbf{0}$
$\bar{d}.\mathbf{0}$
$\bar{d}$

Figure 1.12: The process $a.(b.\mathbf{0} + c.\bar{d}.\mathbf{0})$.

For instance, consider the process $a.(b.\mathbf{0} + c.\bar{d}.\mathbf{0})$ in Figure 1.12. By the axiom (pref) we have that $a.(b.\mathbf{0} + c.\bar{d}.\mathbf{0})$ performs an a-labelled transition to $b.\mathbf{0} + c.\bar{d}.\mathbf{0}$. We derive from the axiom (pref) and the two symmetric rules for sum that:

$$\dfrac{\dfrac{}{b.\mathbf{0} \xrightarrow{b} \mathbf{0}}\ (\text{pref})}{b.\mathbf{0} + c.\bar{d}.\mathbf{0} \xrightarrow{b} \mathbf{0}}\ (\text{sumL}) \qquad \dfrac{\dfrac{}{c.\bar{d}.\mathbf{0} \xrightarrow{c} \mathbf{0}}\ (\text{pref})}{b.\mathbf{0} + c.\bar{d}.\mathbf{0} \xrightarrow{b} \bar{d}.\mathbf{0}}\ (\text{sumR})$$

Finally, by the axiom (pref) we have that $\bar{d}.\mathbf{0} \xrightarrow{\bar{d}} \mathbf{0}$.

Figure 1.13 represents the LTSs associated to some other CCS-processes.

CCS provides an elegant axiomatization of bisimilarity on finite processes (Sangiorgi 2012a) and finite-state processes (Milner 1989).[28] The soundness of these axiomatizations follows

[27] For technical reasons, we restrict our attention to processes where both sums and constants are guarded. A sum is guarded if the summed processes are either processes of the form $\alpha.P$ or guarded sums themselves. So, CCS processes can be alternatively described by the following grammar:

$$P ::= \sum_i \alpha.P_i \;\Big|\; P_1 \,|\, P_2 \;\Big|\; (\nu a)P \;\Big|\; K$$

where the empty sum denotes the process $\mathbf{0}$ and the sum of a single element is a prefixed process $\alpha.P$. An occurrence of a constant in a process is guarded when it occurs within a subterm of the form $\alpha.P$. A constant $A = P$ is guarded if any occurrence of a constant (A included) in P is guarded.
For instance, the defining equations of unguarded constants do not always have an unique solution (Milner 1983) and the process represented by a guarded constant is image-finite (Baeten 1990).

[28] Bisimilarity is undecidable on arbitrary processes and it has no axiomatizations (Aceto, Ingólfsdóttir, and Srba 2012).

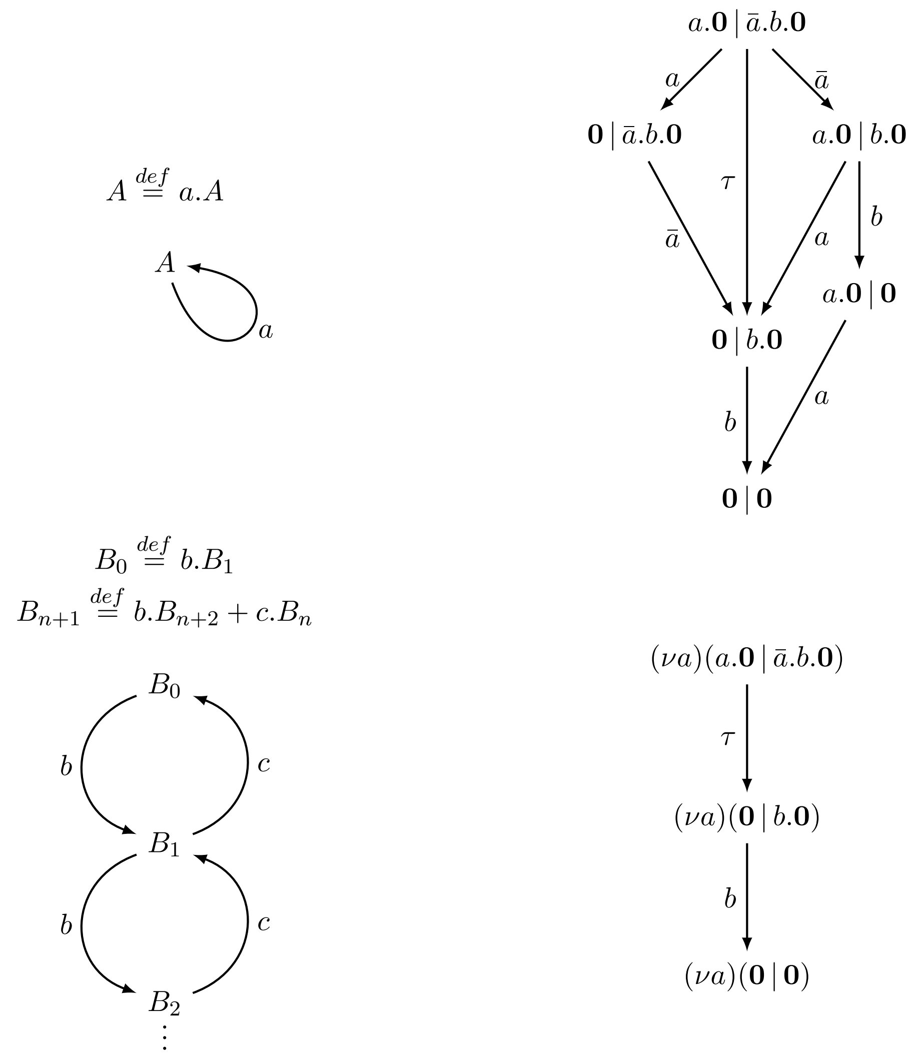
$A \overset{def}{=} a.A$
A
a
$B_0 \overset{def}{=} b.B_1$
$B_{n+1} \overset{def}{=} b.B_{n+2} + c.B_n$
B_0
b
c
B_1
b
c
B_2
$\vdots$
$a.\mathbf{0} \mid \bar{a}.b.\mathbf{0}$
a
$\bar{a}$
τ
$\mathbf{0} \mid \bar{a}.b.\mathbf{0}$
$a.\mathbf{0} \mid b.\mathbf{0}$
$\bar{a}$
a
b
$a.\mathbf{0} \mid \mathbf{0}$
$\mathbf{0} \mid b.\mathbf{0}$
b
a
$\mathbf{0} \mid \mathbf{0}$
$(\nu a)(a.\mathbf{0} \mid \bar{a}.b.\mathbf{0})$
τ
$(\nu a)(\mathbf{0} \mid b.\mathbf{0})$
b
$(\nu a)(\mathbf{0} \mid \mathbf{0})$

Figure 1.13: Some CCS processes.

from the fact that bisimilarity is a congruence with respect to the operators of CCS. Moreover, CCS is a Turing-complete language (Milner 1989, Gorrieri 2013): for every Turing Machine there is a CCS-term encoding it.[29]

[29] In this thesis we will not be concerned with these results, so we omit the proofs and refer the reader to the cited works.

Chapter 2

Probabilistic processes

This chapter analyzes a refinement of Labelled Transition Systems: probabilistic Labelled Transition Systems. In Section 2.2 we define probabilistic bisimilarity, a form of bisimilarity taking into account a state's probability of reaching other states.
We focus our attention on a subset of probabilistic processes, the so-called probabilistic reactive processes. Probabilistic Modal Logic (Section 2.2.3) allows us to capture probabilistic bisimilarity on probabilistic reactive processes.
We conclude this chapter by illustrating a testing scenario for probabilistic processes due to Kim G. Larsen and Arne Skou (Larsen and Skou 1991).

2.1 Probabilistic Labelled Transition Systems

Probabilistic Labelled Transition Systems are obtained by adding to Labelled Transition Systems a quantitative information, namely the probability of performing certain transitions. In the literature, a great variety of probabilistic refinements of Labelled Transition Systems has been proposed.[30]
The transition relation of a Labelled Transition System was defined in the previous chapter as a binary relation on states; in this section, we define Labelled Transition Systems whose transitions are from states to probability distributions on states. Before providing the formal definition of a probabilistic Labelled Transition System, we recall a few preliminary notions about probability distributions.

Let S be a set and let Δ be a function from S to $[0, 1]$. The support of Δ is the set $\lceil \Delta \rceil = \{s \in S| \ \Delta(s) > 0\}$. We say that Δ is a (discrete probability) distribution on S if $\sum_{s \in \lceil \Delta \rceil} \Delta(s) = 1$. We use $\mathcal{D}(S)$ to denote the set of all distributions on S. Given

[30] We refer the reader to (Sokolova and Vink 2004) for a detailed comparison of this variety of probabilistic structures. In (Bernardo, De Nicola, and Loreti 2013a) the ULTraS model, a uniform framework for modeling various refinements of Labelled Transition Systems, is introduced.

a distribution Δ on S and a set $S' \subseteq S$, we extend Δ to the powerset of S by letting $\Delta(S') = \sum\{\Delta(s) |\ s \in S'\}$. Let $\Delta_1, ..., \Delta_n$ be distributions on S and let $p_1, ..., p_n$ be a collection of positive real numbers such that $\sum_{i=1}^{n} p_i = 1$. We define the distribution $\sum_{i=1}^{n} p_i \times \Delta_i$ as follows:

$$(\sum_{i=1}^{n} p_i \times \Delta_i)(s) = \sum_{i=1}^{n} p_i \times \Delta_i(s).$$

A Dirac distribution is a probability distribution assigning 1 to a single element of S. We write $\overline{s}$ to denote the Dirac probability distribution such that:

$$\overline{s}(s') = \begin{cases} 1 & \text{if } s = s' \\ 0 & \text{else.} \end{cases}$$

We often write probability distributions as sums of probability values applied to Dirac distributions. For instance, the distribution

$$\Delta = \frac{1}{2} \cdot \overline{s'} + \frac{1}{4} \cdot \overline{s''} + \frac{1}{4} \cdot \overline{s'''}$$

behaves as follows:

$$\Delta(s) = \begin{cases} \frac{1}{2} & \text{if } s = s' \\ \frac{1}{4} & \text{if } s = s'' \\ \frac{1}{4} & \text{if } s = s''' \\ 0 & \text{else.} \end{cases}$$

Definition 2.1. A *probabilistic Labelled Transition System* (pLTS) is a tuple $\mathcal{T} = \langle St, \mathscr{A}, \longrightarrow \rangle$ where:

- St is a non-empty set of states or probabilistic processes,

- $\mathscr{A}$ is a set of atomic actions,

- $\longrightarrow \subseteq St \times \mathscr{A} \times \mathcal{D}(St)$ is a labelled transition relation between states and distributions on states.

Analogously to what we did in the previous chapter, we use $P, Q, R, S...$ (and their indexed variants $P_1, P_2 \ldots, R_1, R_2, \ldots$) to range over probabilistic processes and μ (and its indexed variants $\mu_1, \mu_2 \ldots$) to range over $\mathscr{A}$, whose elements are denoted by $a, b, c, \ldots$.
For instance, the PLTS in Figure 2.1 represents the tuple $\mathcal{T} = \langle St, \mathscr{A}, \longrightarrow \rangle$, where:

$$\begin{aligned} St =& \{P, P_1, P_2, P_3, P_4, P_5, P_6\} \\ \mathscr{A} =& \{a, b, c\} \end{aligned}$$

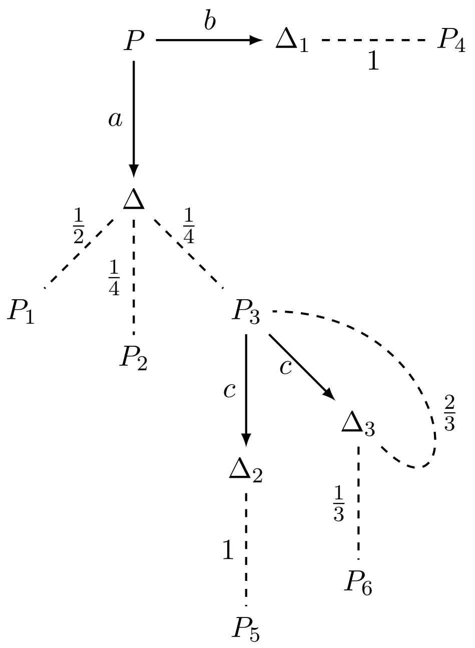

Figure 2.1: A pLTS.

$$\longrightarrow=\{(P,a,\Delta),(P,b,\Delta_1),(P_3,c,\Delta_2),(P_3,c,\Delta_3)\}$$

and the probability distributions $\Delta, \Delta_1, \Delta_2, \Delta_3$ are defined as follows:

$$\Delta = \frac{1}{2}\cdot\overline{P_1} + \frac{1}{4}\cdot\overline{P_2} + \frac{1}{4}\cdot\overline{P_3} \qquad \Delta_1 = \overline{P_4}$$

$$\Delta_2 = \overline{P_5} \qquad \Delta_3 = \frac{1}{3}\cdot\overline{P_6} + \frac{2}{3}\cdot\overline{P_3}$$

Given a pLTS $\mathcal{T} = \langle St, \mathscr{A}, \longrightarrow\rangle$, we write:

- $P \xrightarrow{\mu} \Delta$ whenever $(P,\mu,\Delta) \in\longrightarrow$,
- $P \not\xrightarrow{\mu} \Delta$ if $(P,\mu,\Delta) \not\in\longrightarrow$,
- $P \not\xrightarrow{\mu}$ if $P \not\xrightarrow{\mu} \Delta$ for all distributions Δ,
- $P \not\longrightarrow$ if $P \not\xrightarrow{\mu}$ for all atomic actions μ.

Sometimes we will assume, mainly for technical reasons, that a pLTS satisfies one of the properties listed below. Let $\mathcal{T} = \langle St, \mathscr{A}, \longrightarrow\rangle$ be a pLTS.

$\mathcal{T}$ satisfies the *minimal probability assumption* if there exists a minimal probability value $\pi \in (0,1]$ such that if $P \xrightarrow{\mu} \Delta$ then $\Delta(P') \neq 0$ implies $\Delta(P') \geq \pi$, for all probabilistic processes P, P' and for every μ.

$\mathcal{T}$ satisfies the *minimal deviation assumption* if there exists a minimal probability value $\pi \in (0,1]$ such that if $P \xrightarrow{\mu} \Delta$ then for some $n \in \mathbb{N}$, $\Delta(P') = n{\cdot}\pi$, for all probabilistic processes P, P' and for every μ.

It is easy to check that if a pLTS satisfies the minimal probability assumption then the support of every distribution reachable from a state in the pLTS is finite. The minimal deviation assumption implies the minimal probability assumption: if $P \xrightarrow{\mu} \Delta$ implies that $\Delta(P') = n \cdot \pi$ for some $n \in \mathbb{N}$, for all P, P' and for every μ, then $P \xrightarrow{\mu} \Delta$ and $\Delta(P') \neq 0$ implies that $\Delta(P') \geq \pi$, for all P, P' and for every μ.

Fully nondeterministic processes (i.e. Labelled Transition Systems) are themselves representable as a subclass of probabilistic Labelled Transition Systems, namely the subclass of those probabilistic processes reaching only Dirac distributions.
A probabilistic Labelled Transition Systems can exhibit both external and internal nondeterminism. In the first case, there is a state performing more than one transition, and these transitions have different labels; in the second one, there are multiple transitions with the same label leaving the same state. Reactive probabilistic Labelled Transition Systems compose the subclass of those probabilistic Labelled Transition Systems that only have external nondeterminism.

Definition 2.2. A pLTS $\mathcal{T} = \langle St, \mathscr{A}, \longrightarrow \rangle$ is a *reactive pLTS* if

$$P \xrightarrow{\mu} \Delta_1 \text{ and } P \xrightarrow{\mu} \Delta_2 \text{ implies } \Delta_1 = \Delta_2$$

for all P, μ and $\Delta_1, \Delta_2 \in \mathcal{D}(St)$.

We say that P is a probabilistic process (respectively: a reactive probabilistic process) if P is a state of a pLTS (respectively: a state of a reactive pLTS).

In the field of concurrency theory,[31] reactive probabilistic processes were first studied in (Larsen and Skou 1991) and (van Glabbeek et al. 1990).[32] As we mentioned above, we can look at reactive probabilistic processes as to a first refinement of nondeterministic processes where internal choices are equipped with specific probability values. For instance, consider the fully nondeterministic process V represented in Figure 1.8 and suppose that we come to know that when performing an a-labelled transition the process V becomes one half of the times the process V_1 and one half of the times the process V_2. Then we represent V as the reactive probabilistic process in Figure 2.2.

[31] Reactive probabilistic processes are the counterparts in concurrency theory of structures known in probability theory as reward-free Markov decision processes (Puterman 1994).

[32] This work was extender further in (van Glabbeek, Smolka, and Steffen 1995).

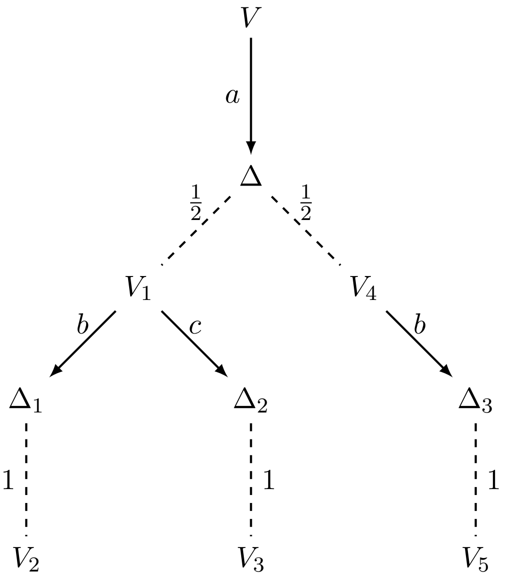

Figure 2.2: A reactive probabilistic process.

2.2 Probabilistic bisimilarity

Different notions of probabilistic bisimilarity appear in the literature.[33] In Section 2.2.1 we define probabilistic bisimulation as presented in (Segala and Lynch 1994, Segala and Lynch 1995) under the name of "strong bisimilarity".[34] As pointed out in (Deng and Du 2011), there are several possible alternative characterizations of this probabilistic bisimilarity. In the following section we use a characterization based on the lifting of binary relations from states to probability distributions on states.

2.2.1 Probabilistic bisimilarity through lifted relations

Definition 2.3. Let $\mathcal{T} = \langle St, \mathscr{A}, \longrightarrow \rangle$ be a pLTS and let $\mathcal{R} \subseteq St \times St$. We lift it to a relation $\overline{\mathcal{R}}$ between distributions on probabilistic processes by letting $\Delta \, \overline{\mathcal{R}} \, \Theta$ whenever there is a finite index set I such that:

1. $\Delta = \sum_{i \in I} p_i \cdot \overline{P_i}$ and $\sum_{i \in I} p_i = 1$,

2. for every $i \in I$ there is a probabilistic process Q_i such that $P_i \, \mathcal{R} \, Q_i$,

[33] We refer the reader to (Hennessy 2012) for a comparison of various definitions of bisimulations in the probabilistic setting. In (Parma and Segala 2007) and (Hermanns et al. 2011), both the characterization through approximants and the logical characterization of probabilistic bisimilarities are analyzed.

[34] In these works, Roberto Segala and Nancy Lynch also define a probabilistic version of simulation equivalence. Differently from what we did with respect to fully nondeterministic processes, we do not examine the spectrum of equivalences for probabilistic processes. This spectrum is studied in (Bernardo, De Nicola, and Loreti 2013d) and (Bernardo, De Nicola, and Loreti 2013b).

3. $\Theta = \sum_{i\in I} p_i \cdot \overline{Q_i}$.

We say that the relation $\overline{\mathcal{R}}$ is the *lifting* of $\mathcal{R}$.

Definition 2.4. Let $\mathcal{T} = \langle St, \mathscr{A}, \longrightarrow\rangle$ be a pLTS. A relation $\mathcal{R} \subseteq St \times St$ is a *probabilistic bisimulation* if whenever $P\,\mathcal{R}\,Q$:

- for every $\mu \in \mathscr{A}$ and for every $\Delta \in \mathcal{D}(St)$, if $P \xrightarrow{\mu} \Delta$ then $Q \xrightarrow{\mu} \Theta$ and $\Delta\,\overline{\mathcal{R}}\,\Theta$, for some distribution $\Theta \in \mathcal{D}(St)$,

- for every $\mu \in \mathscr{A}$, $\Theta \in \mathcal{D}(St)$, if $Q \xrightarrow{\mu} \Theta$ then $P \xrightarrow{\mu} \Delta$ and $\Delta\,\overline{\mathcal{R}}\,\Theta$, for some distribution $\Delta \in \mathcal{D}(St)$.

P and Q are probabilistically bisimilar (written $P \sim Q$) if there exists a probabilistic bisimulation $\mathcal{R}$ such that $P\,\mathcal{R}\,Q$.

The following lemma is from (Deng and Du 2007) and it is useful for the proof of Theorem 2.6. It states that the composition of the liftings of two binary relations is included in the lifting of the composition of the two relations.

Lemma 2.5. *Let S be a set and let $\mathcal{R}_1$ and $\mathcal{R}_2$ be binary relations on S. If $\Delta\,\overline{\mathcal{R}_1}\,\Delta_1$ and $\Delta_1\,\overline{\mathcal{R}}\,\Delta_2$ then $\Delta\,\overline{\mathcal{R}_1 \circ \mathcal{R}_2}\,\Delta_2$, for all distributions $\Delta, \Delta_1, \Delta_2$ on S.*

Proof. If $\Delta\,\overline{\mathcal{R}_1}\,\Delta_1$ and $\Delta_1\,\overline{\mathcal{R}}\,\Delta_2$ then there are two finite index sets I, J such that:

$$\Delta = \sum_{i\in I} p_i \cdot \overline{s_i} \qquad \text{and} \qquad \Delta_1 = \sum_{i\in I} p_i \cdot \overline{s'_i}\ , \qquad \text{where } s_i\,\mathcal{R}_1 s'_i \text{ for all } i \in I$$

$$\Delta_1 = \sum_{j\in J} q_j \cdot \overline{t'_j} \qquad \text{and} \qquad \Delta_2 = \sum_{j\in J} q_j \cdot \overline{t''_j}\ , \qquad \text{where } t'_j\,\mathcal{R}_2 t''_j \text{ for all } j \in J$$

Let $J_i = \{j \in J \mid t'_j = s'_i\}$ and $I_j = \{i \in I \mid s'_i = t'_j\}$ for every $i \in I$ and for every $j \in J$, respectively. The sets $\{(i,j) \mid i \in I, j \in J_i\}$ and $\{(i,j) \mid j \in J, i \in I_j\}$ coincide and we have that:

$$(*) \qquad \Delta_1(s'_i) = \sum_{j\in J_i} q_j \qquad \text{and} \qquad \Delta_1(t'_j) = \sum_{i\in I_j} p_i$$

Thus,

$$\begin{aligned}
\Delta =& \sum_{i\in I} p_i \cdot \overline{s_i} = \\
=& \sum_{i\in I} p_i \cdot \frac{\sum_{j\in J_i} q_j}{\Delta_1(s'_i)} \cdot \overline{s_i} = && \text{(by (*))} \\
=& \sum_{i\in I}\sum_{j\in J_i} \frac{p_i \cdot q_j}{\Delta_1(s'_i)} \cdot \overline{s_i} =
\end{aligned}$$

$$= \sum_{\{(i,j)|\, i\in I, j\in J_i\}} \frac{p_i \cdot q_j}{\Delta_1(s'_i)} \cdot \overline{s_i}$$

and

$$\begin{aligned}
\Delta_2 &= \sum_{j\in J} q_j \cdot \overline{t''_j} = \\
&= \sum_{j\in J} q_j \cdot \frac{\sum_{i\in I_j} p_i}{\Delta_1(t'_j)} \cdot \overline{t''_j} = && \text{(by (*))} \\
&= \sum_{j\in J} \sum_{i\in I_j} \frac{p_i \cdot q_j}{\Delta_1(t'_j)} \cdot \overline{t''_j} = \\
&= \sum_{\{(i,j)|\, j\in J, i\in I_j\}} \frac{p_i \cdot q_j}{\Delta_1(t'_j)} \cdot \overline{t''_j} = \\
&= \sum_{\{(i,j)|\, i\in I, j\in J_i\}} \frac{p_i \cdot q_j}{\Delta_1(t'_j)} \cdot \overline{t''_j} = \\
&= \sum_{\{(i,j)|\, i\in I, j\in J_i\}} \frac{p_i \cdot q_j}{\Delta_1(s'_i)} \cdot \overline{t''_j}.
\end{aligned}$$

By definition, for every $(i,j) \in \{(i,j)|\, i \in I, j \in J_i\}$ we have that $s_i \mathcal{R}_1 s'_i = t'_j \mathcal{R}_2 t''_j$, so $\Delta \overline{\mathcal{R}_1 \circ \mathcal{R}_2} \Delta_2$. □

We can now prove that probabilistic bisimilarity actually is an equivalence relation on probabilistic processes. Moreover, probabilistic bisimilarity is itself a probabilistic bisimulation and it is the largest probabilistic bisimulation as well, since probabilistic bisimilarity includes all probabilistic bisimulations.

Theorem 2.6. *Let $\mathcal{T} = \langle St, \mathscr{A}, \longrightarrow \rangle$ be a pLTS.*

(1) $\sim$ *is a an equivalence relation,*

(2) $\sim$ *is the largest probabilistic bisimulation on St.*

Proof.

(1) We prove that $\sim$ is reflexive, symmetric and transitive.

- The identity relation $\mathcal{I}$ on the processes in $\mathcal{T}$ is a probabilistic bisimulation: if $P \xrightarrow{\mu} \Delta$ and $\Delta = \sum_{i\in I} p_i \cdot \overline{P_i}$ then $P \xrightarrow{\mu} \Delta = \sum_{i\in I} p_i \cdot \overline{P_i}$ and (Δ, Δ) is in the lifting of the relation $\mathcal{I}$, and vice versa.
- The inverse relation $\mathcal{R}^{-1}$ of a probabilistic bisimulation is itself a probabilistic bisimulation.
- Suppose that $P \mathcal{R}_1 P_1$ and $P_1 \mathcal{R}_2 P_2$, where $\mathcal{R}_1$ and $\mathcal{R}_2$ are probabilistic bisimulations. If $P \xrightarrow{\mu} \Delta$ then there is a distribution Δ_1 such that P_1 reaches Δ_1 by

performing a μ-labelled transition and $\Delta\overline{\mathcal{R}_1}\Delta_1$. It follows from $P_1 \xrightarrow{\mu} \Delta_1$ and $P_1\,\mathcal{R}_2 P_2$ that P_2 can perform a μ-labelled transition to a distribution Δ_2 such that $\Delta_1\overline{\mathcal{R}_2}\Delta_2$. By Lemma 2.5, $\Delta\overline{\mathcal{R}_1 \circ \mathcal{R}_2}\Delta_2$, so $\mathcal{R}_1 \circ \mathcal{R}_2$ is a probabilistic bisimulation and the result follows.

(2) Let $P \xrightarrow{\mu} \Delta$ and $P \sim Q$. Then there is a relation $\mathcal{R}$ such that $\mathcal{R}$ is a probabilistic bisimulation and $P\,\mathcal{R}\,Q$ and $\Delta\overline{\mathcal{R}}\Theta$, for some distribution Θ such that $Q \xrightarrow{\mu} \Theta$. If $\Delta\overline{\mathcal{R}}\Theta$ then there is a finite index set I such that $\Delta = \sum_{i\in I} p_i{\cdot}\overline{P_i}$ and $\Theta = \sum_{i\in I} p_i{\cdot}\overline{Q_i}$, where $P_i\,\mathcal{R}\,Q_i$ for all $i \in I$. If $P_i\,\mathcal{R}\,Q_i$ then $P_i \sim Q_i$, so $\Delta\overline{\sim}\Theta$. The second condition in the definition of probabilistic bisimulation is symmetrically satisfied by $\sim$, thus $\sim$ is a probabilistic bisimulation.

Probabilistic bisimilarity is also the largest probabilistic bisimulation, for all probabilistic bisimulations are included in $\sim$.

□

2.2.2 Probabilistic bisimilarity on reactive probabilistic processes

If we restrict our attention to reactive probabilistic processes, it is possible to define probabilistic bisimilarity by considering an equivalence relation from the very beginning and by comparing the equivalence classes induced by this relation. This is the approach that was originally proposed in (Larsen and Skou 1991) for reactive probabilistic Labelled Transition Systems.[35]

Definition 2.7. Let $\mathcal{T} = \langle St, \mathscr{A}, \longrightarrow\rangle$ be a pLTS. An equivalence relation $\mathcal{R} \subseteq St \times St$ is an *ls-probabilistic bisimulation* if whenever $P\,\mathcal{R}\,Q$:

- for every $\mu \in \mathscr{A}$ and for every $\Delta \in \mathcal{D}(St)$, if $P \xrightarrow{\mu} \Delta$ then there is a distribution Θ such that $Q \xrightarrow{\mu} \Theta$ and $\Delta(E) = \Theta(E)$, for every equivalence class E in ${}^{St}\!/_{\mathcal{R}}$.

P and Q are ls-probabilistically bisimilar (written $P \sim_{ls} Q$) if there exists an ls-probabilistic bisimulation $\mathcal{R}$ such that $P\,\mathcal{R}\,Q$.

Theorem 2.8 proves that probabilistic bisimilarity and ls-probabilistic bisimilarity coincide on the class of reactive probabilistic processes.

Theorem 2.8. *Let $\mathcal{T} = \langle St, \mathscr{A}, \longrightarrow\rangle$ be a reactive pLTS. $P \sim Q$ if and only if $P \sim_{ls} Q$.*

Proof.

[35] Due to the lack of internal nondeterminism, probabilistic bisimilarity on reactive probabilistic processes coincides with probabilistic simulation equivalence. This result was proved in (Baier and Kwiatkowska 2000) and (Desharnais et al. 2003).

($\Rightarrow$) We show that $\sim$ is an ls-probabilistic bisimulation. By Theorem 2.6, $\sim$ is an equivalence relation. If $P \xrightarrow{\mu} \Delta$ then there is a Θ such that $Q \xrightarrow{\mu} \Theta$ and $\Delta \overline{\mathcal{R}} \Theta$, i.e. there is a finite index set I such that $\Delta = \sum_{i \in I} p_i \cdot \overline{P_i}$, $\Theta = \sum_{i \in I} p_i \cdot \overline{Q_i}$ and $P_i \mathcal{R} Q_i$ for every $i \in I$.

Let E be an equivalence class in ${}^{St}/_{\sim}$ and let K be the subset of I whose elements are all and only the indexes of processes in $E \cap \lceil \Delta \rceil$. By the hypothesis, $P_i \mathcal{R} Q_i$ for every $i \in I$, hence $\Theta(E) \geq \sum_{k \in K} p_k = \Delta(E)$. Now, suppose that $\Theta(E) > \sum_{k \in K} p_k$. Then there is a $j \in I \setminus K$ such that $Q_j \in E$ and it follows from $P_j \sim Q_j$ that $P_j \in E$, which contradicts the assumption that $\{P_i | \; i \in K\} = E \cap \lceil \Delta \rceil$. Therefore, $\Theta(E) = \sum_{k \in K} p_k = \Delta(E)$.

($\Leftarrow$) We prove that if an equivalence relation $\mathcal{R}$ is an ls-probabilistic bisimulation then $\mathcal{R}$ is a probabilistic bisimulation, i.e. we prove that if $P \xrightarrow{\mu} \Delta$ then there is a Θ such that $Q \xrightarrow{\mu} \Theta$ and $\Delta \overline{\mathcal{R}} \Theta$. The second condition in the definition of $\sim$ follows from the fact that $\mathcal{R}$ is a symmetric relation.

Let $\mathcal{R}$ be an ls-probabilistic bisimulation. Suppose that $P \mathcal{R} Q$ and $P \xrightarrow{\mu} \Delta$. Then there is a Θ such that $Q \xrightarrow{\mu} \Theta$ and for every equivalence class $E \in {}^{St}/_{\mathcal{R}}$, $\Delta(E) = \Theta(E)$. Define the index set

$$I = \{(S,T) | \; S \mathcal{R} T \text{ and } S \in \lceil \Delta \rceil \text{ and } T \in \lceil \Theta \rceil\},$$

which is finite, both $\lceil \Delta \rceil$ and $\lceil \Theta \rceil$ being finite. Let $p_{(s,t)} = \frac{\Delta(S) \cdot \Theta(T)}{\Delta([S])}$; it follows from the hypothesis that for every couple (S,T), $p_{(s,t)} = \frac{\Delta(S) \cdot \Theta(T)}{\Delta([T])}$. For every $(S,T) \in I$, let $l_{(S,T)} = S$ and $r_{(S,T)} = T$. We prove that the following distributions on St:

$$\Delta' = \sum_{(S,T) \in I} p_{(s,t)} \cdot \overline{l_{(S,T)}} \qquad\qquad \Theta' = \sum_{(S,T) \in I} p_{(s,t)} \cdot \overline{r_{(S,T)}}$$

are equivalent to Δ and Θ, respectively. Let $S' \in \lceil \Delta \rceil$. Therefore,

$$\begin{aligned}
\Delta'(S') = (\sum_{(S,T) \in I} \frac{\Delta(S) \cdot \Theta(T)}{\Delta([S])} \cdot \overline{l_{(S,T)}})(S') = \\
\sum_{(S,T) \in I} \frac{\Delta(S) \cdot \Theta(T)}{\Delta([S])} \cdot \overline{l_{(S,T)}}(S') = \\
= \sum_{\{(S,T) \in I | \; S = S'\}} \frac{\Delta(S) \cdot \Theta(T)}{\Delta([S])} = \\
= \sum_{T \in [S'] \cap \lceil \Theta \rceil} \frac{\Delta(S') \cdot \Theta(T)}{\Delta([S'])} = \\
= \Delta(S') \cdot \frac{\sum_{T \in [S'] \cap \lceil \Theta \rceil} \Theta(T)}{\Delta([S'])} =
\end{aligned}$$

$$\begin{aligned}
&= \Delta(S') \cdot \frac{\Theta([S'])}{\Delta([S'])} = \\
&= \Delta(S') \cdot \frac{\Delta([S'])}{\Delta([S'])} = && \text{(Hp)} \\
&= \Delta(S').
\end{aligned}$$

Symmetrically, it holds that $\Theta(T) = \Theta'(T)$ for every T. For every $(S, T) \in I$, $S \mathcal{R} T$ (they belong to the same equivalence class), then $\Delta \overline{\mathcal{R}} \Theta$, since Δ' and Θ' are decompositions of Δ and Θ that satisfy the requirements of Definition 2.3.

□

2.2.3 Probabilistic Modal Logic

Probabilistic Modal Logic is a probabilistic variant of Ready Simulation Logic (Definition 1.37). The diamond operator of Probabilistic Modal Logic is indexed by a probability value: the formula $\langle \mu \rangle_p F$ is true at a state P if and only if there is a μ-labelled transition from P such that the probability of reaching a state satisfying F is greater than or equal to p.

Probabilistic Modal Logic was defined in (Larsen and Skou 1991) in the setting of reactive probabilistic processes, while its interpretation on the whole class of probabilistic processes was recently studied in (Bernardo, De Nicola, and Loreti 2013c).

Definition 2.9. Let $\mathscr{A}$ be a set of atomic actions. The formulas of *Probabilistic Modal Logic (PML)* on $\mathscr{A}$ are defined as follows:

$$F ::= \top \;\Big|\; \neg\mu \;\Big|\; F_1 \wedge F_2 \;\Big|\; F_1 \vee F_2 \;\Big|\; \langle \mu \rangle_p F$$

where $\mu \in \mathscr{A}$ and $p \in [0, 1]$.

Let $\mathcal{T} = \langle St, \mathscr{A}, \longrightarrow \rangle$ be a pLTS and let F be a PML-formula We define by induction on the structure of F when $P \models F$:

$$\begin{array}{lll}
P \models \top & \text{always} & \\
P \models \neg\mu & \text{iff} & \text{P} \overset{\mu}{\not\longrightarrow} \\
P \models F_1 \wedge F_2 & \text{iff} & P \models F_1 \text{ and } P \models F_2 \\
P \models F_1 \vee F_2 & \text{iff} & P \models F_1 \text{ or } P \models F_2 \\
P \models \langle \mu \rangle_p F & \text{iff} & P \xrightarrow{\mu} \Delta \text{ and } \sum\{\Delta(P') | P' \models F\} \geq p, \text{ for some } \Delta.
\end{array}$$

$P \equiv^{PML} Q$ if and only for every PML-formula F, $P \models F$ iff $Q \models F$.

Unlike Hennessy-Milner Logic, the formulas of Probabilistic Modal Logic do not have a negation operation. Theorem 2.11 shows the reason why it is not needed if the minimal probability assumption is satisfied.[36]

Definition 2.10. We define by structural induction on the formulas of PML the dual F^D of a formula F:

$$\top^D = \langle\, \mu \,\rangle_1 \top \wedge \neg\mu \qquad \neg\mu^D = \langle\, \mu \,\rangle_1 \top$$

$$(F_1 \wedge F_2)^D = F_1^D \vee F_2^D \qquad (F_1 \vee F_2)^D = F_1^D \wedge F_2^D$$

$$(\langle\, \mu \,\rangle_p F')^D = \neg\mu \vee \langle\, \mu \,\rangle_{1-(K-1)\cdot\pi} F'^D$$

where K is the natural number obtained by rounding up $\frac{p}{\pi}$ to the nearest whole number.

Theorem 2.11. *Let $\mathcal{T} = \langle St, \mathscr{A}, \longrightarrow\rangle$ be a reactive pLTS satisfying the minimal deviation assumption. For every PML-formula F, $P \models F$ if and only if $P \not\models F^D$.*

Proof. By induction on F. Both the case when $F = \top$ and the case when $F = \neg\mu$ immediately follow from the fact that $P \models \langle\, \mu \,\rangle_1 \top$ if and only if $P \xrightarrow{\mu}$. If F is a conjunction or a disjunction of formulas, the result follows directly from the inductive hypothesis. Now, suppose that $F = \langle\, \mu \,\rangle_p F'$. $P \models F$ if and only if $P \xrightarrow{\mu} \Delta$ and $\sum\{\Delta(P')|\ P' \models F'\} \geq p$. If $P \models F$, then $P \xrightarrow{\mu}$ and by the minimal deviation assumption we have that:

$$\sum\{\Delta(P')|\ P' \not\models F'\} = 1 - \sum\{\Delta(P')|\ P' \models F'\} \leq 1 - K \cdot \pi < 1 - (K-1)\cdot\pi.$$

It follows from the inductive hypothesis that $\sum\{\Delta(P')|\ P' \not\models F'\} = \sum\{\Delta(P')|\ P' \models F'^D\}$, so $P \not\models F'^D$. If $P \not\models F$ then either $P \models \neg\mu$ or $\sum\{\Delta(P')|\ P' \models F'\} < p$. In the latter case, it follows from the minimal deviation assumption and from the inductive hypothesis that:

$$\sum\{\Delta(P')|\ P' \models F'^D\} = \sum\{\Delta(P')|\ P' \not\models F'\} = 1 - \sum\{\Delta(P')|\ P' \models F'\} \geq 1-(K-1)\cdot\pi.$$

Hence, $P \models F^D$ and the result follows.

□

Theorem 2.12 shows that Probabilistic Modal Logic characterizes probabilistic bisimilarity on reactive probabilistic Labelled Transition Systems, provided that the minimal deviation assumption is satisfied.

[36] In (Desharnais, Edalat, and Panangaden 2002), the authors prove that the operator $\neg\mu$ of Probabilistic Modal Logic is not needed in order to characterize probabilistic bisimilarity on reactive probabilistic processes, even if we give the finitary assumptions on the support of the distributions up. These results are achieved by considering Labelled Markov processes whose state space is not necessarily discrete.

Theorem 2.12. *Let $\mathcal{T} = \langle St, \mathscr{A}, \longrightarrow \rangle$ be a reactive probabilistic LTS satisfying the minimal deviation assumption. $P \sim Q$ if and only if $P \equiv^{PML} Q$*

Proof.

($\Rightarrow$) We prove by structural induction on F that $P \sim Q$ implies that for every PML-formula, if $P \models F$ then $Q \models F$. The other direction of this last implication follows from the fact that $\sim$ is an equivalence relation (Theorem 2.6(1)). Suppose that $P \sim Q$. We only prove the new inductive cases with respect to the Hennessy-Milner Theorem (Theorem 1.12).

($F = \neg\mu$) Suppose that $Q \not\models F$. We have that $Q \xrightarrow{\mu}$ and it follows from $P \sim Q$ that $P \xrightarrow{\mu}$. Hence, $P \not\models F$.

($F = \langle\, \mu \,\rangle_p F'$) $P \models F$ implies that there exists a Δ such that $P \xrightarrow{\mu} \Delta$ and $\sum\{\Delta(P')|P' \models F\} \geq p$. $P \sim Q$, so there is a Θ such that $\Delta \overline{\sim} \Theta$, i.e. there is a finite index set I such that $\Delta = \sum_{i \in I} p_i \cdot \overline{P_i}$, $\Theta = \sum_{i \in I} p_i \cdot \overline{Q_i}$ and $P_i \sim Q_i$ for every $i \in I$. Let K be the subset of I such that $\{P_i|\; i \in K\} = \{P'|P' \models F\}$. We have that $\sum_{k \in K} p_k = \sum\{\Delta(P_i)|\; i \in K\} = \sum\{\Delta(P')|P' \models F\}$. It follows from the inductive hypothesis that for every $i \in K$, $Q_i \models F$, hence $\sum\{\Theta(Q')|Q' \models F\} \geq \sum_{k \in K} p_k \geq p$. Therefore, $Q \models F$.

($\Leftarrow$) We prove that $\equiv^{PML}$ is a probabilistic bisimulation, which implies the result. Let $P \equiv^{PML} Q$ and $P \xrightarrow{\mu} \Delta$. Then there is a Θ such that $Q \xrightarrow{\mu} \Theta$, since $P \not\models \neg\mu$ and since it follows from $P \equiv^{PML} Q$ that $Q \not\models \neg\mu$. Let $[S]$ be an equivalence class in ${}^{St}/_{\equiv^{PML}}$. Without any loss of generality, we can suppose that:

$\lceil\Delta\rceil = \{P_1, \ldots, P_k, P_{k+1}, \ldots, P_n\}$, where $P_1, \ldots, P_k \in [S]$ and $P_{k+1}, \ldots, P_n \notin [S]$,

$\lceil\Theta\rceil = \{Q_1, \ldots, Q_j, Q_{j+1}, \ldots, Q_m\}$, where $Q_1, \ldots, Q_j \in [S]$ and $Q_{j+1}, \ldots, Q_m \notin [S]$.

It follows from Theorem 2.11 that for every i from $j+1$ to m there is a PML-formula F_i such that $Q_i \not\models F_i$ and $S' \models F_i$ for all $S' \in [S]$. Thus, $P \models \langle\, \mu \,\rangle_{\Delta([S])} (F_{j+1} \wedge \ldots \wedge F_m)$, which implies that $Q \models \langle\, \mu \,\rangle_{\Delta([S])} (F_{j+1} \wedge \ldots \wedge F_m)$ by the hypothesis that $P \equiv^{PML} Q$. Suppose that $\Theta([S]) < \Delta([S])$. For every i from $j+1$ to m, $Q_i \not\models F_{j+1} \wedge \ldots \wedge F_m$, so $Q \not\models \langle\, \mu \,\rangle_{\Delta([S])} (F_{j+1} \wedge \ldots \wedge F_m)$, which leads to a contradiction. Therefore, $\Theta([S]) \geq \Delta([S])$.
$\equiv^{PML}$ is an equivalence relation is an equivalence relation, thus we can repeat the argument symmetrically and we can conclude that $\Theta([S]) = \Delta([S])$ for every equivalence class $[S] \in {}^{St}/_{\equiv^{PML}}$. It follows from Theorem 2.8 that $\equiv^{PML}$ is a probabilistic bisimulation.

□

2.3 Larsen & Skou's testing theory

A further alternative characterization of probabilistic bisimilarity on probabilistic reactive processes was proposed in (Larsen and Skou 1991). In this work, Kim G. Larsen and Arne Skou define a set of tests and apply them to reactive probabilistic processes. These tests have the capability of deciding whether a process can or cannot perform an atomic action and of executing multiple experiments on the same process.

Definition 2.13. Given a set $\mathscr{A}$ of atomic actions, the tests of the testing language $\mathbf{T}^{\mathsf{LS}}$ are defined as follows:

$$t ::= \omega \;\Big|\; \mu.t \;\Big|\; (t_1, ..., t_n)$$

where $\mu \in \mathscr{A}$. Every test t has an inductively defined set of observations O_t:

$$\begin{aligned} O_\omega &= \{1_\omega\} \\ O_{\mu.t} &= \{0_\mu\} \cup \{1_\mu : e |\, e \in O_t\} \\ O_{(t_1,...,t_n)} &= O_{t_1} \times ... \times O_{t_n}. \end{aligned}$$

Let P be a reactive probabilistic process, t a test in $\mathbf{T}^{\mathsf{LS}}$ and $e \in O_t$. The probability $\mathcal{P}_{t,P}(e)$ of observing e when executing the test t on P is defined by structural induction on t:

$$\begin{aligned} &\mathcal{P}_{\omega,P}(1_\omega) = 1 \\ &\mathcal{P}_{\mu.t,P}(0_\mu) = \begin{cases} 1 & \text{if } P \overset{\mu}{\not\longrightarrow} \\ 0 & \text{else} \end{cases} \\ &\mathcal{P}_{\mu.t,P}(1_\mu : e) = \begin{cases} 0 & \text{if } P \overset{\mu}{\not\longrightarrow} \\ \sum_{P' \in \lceil \Delta \rceil} \Delta(P') \cdot \mathcal{P}_{t,P'}(e) & \text{else (where } P \xrightarrow{\mu} \Delta) \end{cases} \\ &\mathcal{P}_{(t_1,...,t_n),P}(e_1, ..., e_n) = \prod_{i=1}^{n} \mathcal{P}_{t_i,P}(e_i) \end{aligned}$$

where $e \in O_t$ and $e_i \in O_i$ for all $i \in \{1, \ldots, n\}$.

For any set $E \subseteq O_t$ of elements of the set O_t of observations on t, we let $\mathcal{P}_{t,P}(E) = \sum_{e \in E} \mathcal{P}_{t,P}(e)$. As a special case we have that $\mathcal{P}_{t,P}(\emptyset) = 0$ for all tests and for all processes. We let E^c denote the complementary set of E in O_t (that is, $E^c = \{e \in O_t |\, e \notin E\}$) and we let $(t)^n$ denote the test:

$$\underbrace{(t, \ldots, t)}_{n\text{-times}}$$

For every test t and for every reactive probabilistic process P, the function $\mathcal{P}_{t,P} : O_t \to [0, 1]$ is a probability distribution on O_t. In fact, it is easy to check that $\mathcal{P}_{t,P}(O_t) = 1$.

2.3.1 Testability of PML-formulas

A property on the states of a reactive probabilistic LTS is ls-testable if there is a Larsen and Skou's test t and there is a set of observations E such that the probability of doing an observation in E when executing the test t on a process is arbitrarily high if the process enjoys the property and it is arbitrarily low if the process does not.

Definition 2.14. Let $\mathcal{T} = \langle St, \mathscr{A}, \longrightarrow\rangle$ be a reactive pLTS. A property ϕ is *ls-testable* if for any $\delta > 0$ there exists a test t_ϕ in $\mathbf{T}^{\mathsf{LS}}$ and there exists a set E_ϕ of observations in O_{t_ϕ} such that for every process P it holds that:

1. if $P \in \phi$ then $\mathcal{P}_{t_\phi,P}(E_\phi) \geq 1 - \delta$,

2. if $P \notin \phi$ then $\mathcal{P}_{t_\phi,P}(E_\phi) \leq \delta$,

The positive real number δ is the *level of significance.*
A class of properties Φ is testable if all the properties in Φ are testable.

The class of properties we are interested in is the class of those sets consisting of all and only the processes satisfying a formula F, for every formula F of Probabilistic Modal Logic. In what follows, we will say that a PML-formula F is testable if the set of processes satisfying F is testable.
The following lemma establishes some useful properties of the subsets of the set of observations of a test.

Lemma 2.15. *Let $\mathcal{T} = \langle St, \mathscr{A}, \longrightarrow\rangle$ be a reactive pLTS. For every test t in $\mathbf{T}^{\mathsf{LS}}$ and for every P, the probability distribution $\mathcal{P}_{t,P}$ on the set of observations O_t has the following properties.*

1. *For every $E_t \subseteq O_t$,*

$$\mathcal{P}_{\mu.t,P}(1_\mu : E_t) = \begin{cases} 0 & \textit{if } P \overset{\mu}{\not\longrightarrow} \\ \sum_{P' \in \lceil \Delta \rceil} \Delta(P') \cdot \mathcal{P}_{t,P'}(E_t) & \textit{else (where } P \overset{\mu}{\longrightarrow} \Delta) \end{cases}$$

where $1_\mu : E_t = \{1_\mu : e |\, e \in E_t\}$.

2. *For every $n \in \mathbb{N}$ and $E_{t_1} \subseteq O_{t_1}, \ldots, E_{t_n} \subseteq O_{t_n}$,*

$$\mathcal{P}_{(t_1,\ldots,t_n),P}(E_1 \times \ldots \times E_n) = \prod_{i=1}^{n} \mathcal{P}_{t_i,P}(E_i).$$

3. *For every $E_t \subseteq O_t$, $\mathcal{P}_{t,P}(E_t^c) = 1 - \mathcal{P}_{t,P}(E_t)$.*

Proof.

1. If $P \stackrel{\mu}{\not\longrightarrow}$ then trivially $\mathcal{P}_{\mu.t,P}(1_\mu : E_t) = 0$. Suppose that $P \xrightarrow{\mu} \Delta$. Then:

$$\begin{aligned}\mathcal{P}_{\mu.t,P}(1_\mu : E_t) &= \sum_{e \in E_t} \mathcal{P}_{\mu.t,P}(1_\mu : e) = \\ &= \sum_{e \in E_t} \sum_{P' \in \lceil \Delta \rceil} \Delta(P') \cdot \mathcal{P}_{t,P'}(e) = \\ &= \sum_{P' \in \lceil \Delta \rceil} \Delta(P') \cdot \sum_{e \in E_t} \mathcal{P}_{t,P'}(e) = \\ &= \sum_{P' \in \lceil \Delta \rceil} \Delta(P') \cdot \mathcal{P}_{t,P'}(E_t).\end{aligned}$$

2. It holds that:

$$\begin{aligned}\mathcal{P}_{(t_1,\dots,t_n),P}(E_1 \times \dots \times E_n) &= \sum_{(e_1,\dots,e_n) \in E_1 \times \dots \times E_n} \prod_{i=1}^{n} \mathcal{P}_{t_i,P}(e_i) = \\ &= \sum_{e_1 \in E_1} \sum_{e_2 \in E_2} \cdots \sum_{e_n \in E_n} \prod_{i=1}^{n} \mathcal{P}_{t_i,P}(e_i) = \\ &= \sum_{e_1 \in E_1} \sum_{e_2 \in E_2} \cdots \prod_{i=1}^{n} \sum_{e_n \in E_n} \mathcal{P}_{t_i,P}(e_i) = \\ &\ \vdots \\ &= \prod_{i=1}^{n} \sum_{e_1 \in E_1} \sum_{e_2 \in E_2} \cdots \sum_{e_n \in E_n} \mathcal{P}_{t_i,P}(e_i) = \\ &= \prod_{i=1}^{n} \sum_{(e_1,\dots,e_n) \in E_1 \times \dots \times E_n} \mathcal{P}_{t_i,P}(e_i) = \\ &= \prod_{i=1}^{n} \mathcal{P}_{t_i,P}(E_i).\end{aligned}$$

3. $\mathcal{P}_{t,P}$ is a probability distribution on O_t, so $\mathcal{P}_{t,P}(E) + \mathcal{P}_{t,P}(E^c) = \mathcal{P}_{t,P}(O_t) = 1$ and the result follows.

□

We can now prove that the formulas of Probabilistic Modal Logic are testable on the class of reactive probabilistic processes satisfying the minimal deviation assumption. The proof of Theorem 2.16 involves some definitions and results from probability theory, which are presented e.g. in (Bertsekas and Tsitsiklis 2008) and (DeGroot and Schervish 2012).

Theorem 2.16. *Let $\mathcal{T} = \langle St, \mathscr{A}, \longrightarrow \rangle$ be a reactive pLTS satisfying the minimal deviation assumption. The formulas of Probabilistic Modal Logic are testable.*

Proof. The proof is by structural induction on PML-formulas.

($F = \top$) Let $t_\top = \omega$ and $E_\top = \{1_\omega\}$. It always holds that $P \models \top$ and $\mathcal{P}_{\omega,P}(1_\omega) = 1 \geq 1 - \delta$ for every $\delta > 0$.

($\neg\mu$) Let $t_{\neg\mu} = \mu.t$ and $E_{\neg\mu} = \{0_\mu\}$. If $P \models \neg\mu$ then $P \overset{\mu}{\not\rightarrow}$ and $\mathcal{P}_{\mu.t,P}(0_\mu) = 1$. If $P \not\models \neg\mu$ then $P \xrightarrow{\mu}$ and $\mathcal{P}_{\mu.t,P}(0_\mu) = 0$.

($F = F_1 \wedge F_2$) By the inductive hypothesis, for any level of significance δ_1 there exists a test t_{F_1} and a set $E_{F_1} \subseteq O_{t_{F_1}}$ satisfying the definition of testability for F_1. Analogously, for any level of significance δ_2 there exists a test t_{F_2} and a set $E_{F_2} \subseteq O_{t_{F_2}}$ satisfying the definition of testability for F_2. If $P \models F_1 \wedge F_2$ then:

$$\begin{aligned}
\mathcal{P}_{(t_{F_1},t_{F_2}),P}(E_{F_1} \times E_{F_2}) &= \mathcal{P}_{t_{F_1},P}(E_{F_1}) \cdot \mathcal{P}_{t_{F_2}}(E_{F_2}) && \text{(by Lemma 2.15(2))}\\
&\geq (1 - \delta_{F_1}) \cdot (1 - \delta_{F_2}) && \text{(by HI)}\\
&\geq 1 - (\delta_{F_1} + \delta_{F_2}).
\end{aligned}$$

If $P \not\models F_1 \wedge F_2$ then it follows from the inductive hypothesis that $\mathcal{P}_{t_{F_1},P}(E_{F_1}) \leq \delta_1$ or $\mathcal{P}_{t_{F_2},P}(E_{F_2}) \leq \delta_2$, thus:

$$\begin{aligned}
&\mathcal{P}_{(t_{F_1},t_{F_2}),P}(E_{F_1} \times E_{F_2}) &&= \mathcal{P}_{t_{F_1},P}(E_{F_1}) \cdot \mathcal{P}_{t_{F_2}}(E_{F_2}) \leq \mathcal{P}_{t_{F_1},P}(E_{F_1}) \leq \delta_1\\
\text{or}\quad &\mathcal{P}_{(t_{F_1},t_{F_2}),P}(E_{F_1} \times E_{F_2}) &&= \mathcal{P}_{t_{F_1},P}(E_{F_1}) \cdot \mathcal{P}_{t_{F_2}}(E_{F_2}) \leq \mathcal{P}_{t_{F_2},P}(E_{F_2}) \leq \delta_2.
\end{aligned}$$

Hence, $\mathcal{P}_{(t_{F_1},t_{F_2}),P}(E_{F_1} \times E_{F_2}) \leq \delta_1 + \delta_2$.
For a given $\delta > 0$, let δ_1 and δ_2 be positive real numbers such that $\delta = \delta_1 + \delta_2$. We proved above that there are a test $t_{F_1 \wedge F_{F_2}} = (t_{F_1}, t_{F_2})$ and an observation set $E_{F_1 \wedge F_2} = E_{F_1} \times E_{F_2}$ such that:

if $P \models F_1 \wedge F_2$ then $\mathcal{P}_{t_{F_1 \wedge F_{F_2}},P}(E_{F_1 \wedge F_2}) \geq 1 - \delta$,

if $P \not\models F_1 \wedge F_2$ then $\mathcal{P}_{t_{F_1 \wedge F_{F_2}},P}(E_{F_1 \wedge F_2}) \leq \delta$.

($F = F_1 \vee F_2$) As in the previous case, it follows from the inductive hypothesis that for every pair of $\delta_1, \delta_2 > 0$ there are tests t_{F_1}, t_{F_2} and observation sets E_{F_1}, E_{F_2} satisfying the conditions of testability for the PML-formulas F_1 and F_2, respectively. If $P \models F_1$ and $P \models F_2$ then, as in the previous case, we have that:

$$(a) \qquad \mathcal{P}_{(t_{F_1},t_{F_2}),P}(E_{F_1} \times E_{F_2}) \geq (1 - \delta_{F_1}) \cdot (1 - \delta_{F_2}).$$

If $P \models F_1$ and $P \not\models F_2$ then:

$$(b) \qquad \mathcal{P}_{(t_{F_1},t_{F_2}),P}(E_{F_1} \times E^{\mathrm{c}}_{F_2}) = \mathcal{P}_{t_{F_1},P}(E_{F_1}) \cdot \mathcal{P}_{t_{F_2}}(E^{\mathrm{c}}_{F_2}) \qquad \text{(by Lemma 2.15(2))}$$

$$\begin{aligned} &= \mathcal{P}_{t_{F_1},P}(E_{F_1}) \cdot (1 - \mathcal{P}_{t_{F_2}}(E_{F_2})) \quad &\text{(by Lemma 2.15(3))}\\ &\geq (1-\delta_{F_1}) \cdot (1-\delta_{F_2}) &\text{(by HI).} \end{aligned}$$

Symmetrically, if $P \not\models F_1$ and $P \models F_2$ then:

$$(c) \qquad \mathcal{P}_{(t_{F_1},t_{F_2}),P}(E^c_{F_1} \times E_{F_2}) \geq (1-\delta_{F_1}) \cdot (1-\delta_{F_2}).$$

By considering the union of the sets of observations in $(a), (b)$ and (c), we derive that $P \models F_1 \vee F_2$ implies that:

$$\begin{aligned} &\mathcal{P}_{(t_{F_1},t_{F_2}),P}(E_{F_1} \times E_{F_2} \cup E_{F_1} \times E^c_{F_2} \cup E^c_{F_1} \times E_{F_2})\\ &\quad = \mathcal{P}_{(t_{F_1},t_{F_2}),P}(E_{F_1} \times E_{F_2}) + \mathcal{P}_{(t_{F_1},t_{F_2}),P}(E_{F_1} \times E^c_{F_2}) + \mathcal{P}_{(t_{F_1},t_{F_2}),P}(E^c_{F_1} \times E_{F_2})\\ &\quad \geq (1-\delta_{F_1}) \cdot (1-\delta_{F_2})\\ &\quad \geq 1 - (\delta_1 + \delta_2). \end{aligned}$$

Finally, if $P \not\models F_1 \wedge F_2$ then:

$$\begin{aligned} &\mathcal{P}_{(t_{F_1},t_{F_2}),P}(E_{F_1} \times E_{F_2} \cup E_{F_1} \times E^c_{F_2} \cup E^c_{F_1} \times E_{F_2})\\ &\quad = \mathcal{P}_{(t_{F_1},t_{F_2}),P}(E_{F_1} \times E_{F_2}) + \mathcal{P}_{(t_{F_1},t_{F_2}),P}(E_{F_1} \times E^c_{F_2}) + \mathcal{P}_{(t_{F_1},t_{F_2}),P}(E^c_{F_1} \times E_{F_2})\\ &\quad = \mathcal{P}_{t_{F_1},P}(E_{F_1}) + \mathcal{P}_{t_{F_2}}(E_{F_2}) - \mathcal{P}_{t_{F_1},P}(E_{F_1}) \cdot \mathcal{P}_{t_{F_2}}(E_{F_2})\\ &\quad \leq \mathcal{P}_{t_{F_1},P}(E_{F_1}) + \mathcal{P}_{t_{F_2}}(E_{F_2})\\ &\quad \leq \delta_1 + \delta_2 \qquad \text{(by HI).} \end{aligned}$$

For any $\delta > 0$, if $\delta = \delta_1 + \delta_2$ then the test $t_{F_1 \wedge F_{F_2}} = (t_{F_1}, t_{F_2})$ and the observation set

$$E_{F_1 \vee F_2} = \{(e_1, e_2) | \, e_1 \in E_1 \vee e_2 \in E_2\} = E_{F_1} \times E_{F_2} \cup E_{F_1} \times E^c_{F_2} \cup E^c_{F_1} \times E_{F_2}$$

satisfy the definition of testability for the PML-formula $F_1 \vee F_2$.

($\langle \mu \rangle_p F$) By the inductive hypothesis on F we have that for every level of significance δ_F there is a test t_F and a set of observations E_F satisfying the definition of testability for any process P. Consider the test $\mu.t_F$ and the set of observations $1_\mu : E_F = \{1_\mu : e | \, e \in E_F\}$.

If $\langle \mu \rangle_p F$ is true at P then there is a probability distribution Δ such that $P \xrightarrow{\mu} \Delta$ and $\Delta(\{P' | \, P' \models F\}) \geq p$. It follows from the minimal deviation assumption that $\Delta(\{P' | \, P' \models F\}) = m \cdot \pi$ for some $m \in \mathbb{N}$ such that $m \geq K$, where K is the natural number obtained by rounding up $\frac{p}{\pi}$ to the nearest whole number. We have that:

$$\mathcal{P}_{\mu.t_F,P}(1_\mu : E_F) =$$

$$
\begin{aligned}
&= \sum_{P' \in \lceil \Delta \rceil} \Delta(P') \cdot \mathcal{P}_{t_F,P'}(E_F) && \text{(by Lemma 2.15(1))}\\
&= \sum_{\{P' \in \lceil \Delta \rceil \mid P' \models F\}} \Delta(P') \cdot \mathcal{P}_{t_F,P'}(E_F) + \sum_{\{P' \in \lceil \Delta \rceil \mid P' \not\models F\}} \Delta(P') \cdot \mathcal{P}_{t_F,P'}(E_F)\\
&\geq \sum_{\{P' \in \lceil \Delta \rceil \mid P' \models F\}} \Delta(P') \cdot \mathcal{P}_{t_F,P'}(E_F)\\
&\geq \sum_{\{P' \in \lceil \Delta \rceil \mid P' \models F\}} \Delta(P') \cdot (1 - \delta_F) && \text{(by HI)}\\
&\geq K \cdot \pi \cdot (1 - \delta_F)\\
&\overset{def}{=} \gamma.
\end{aligned}
$$

If $P \not\models F$ and $P \xrightarrow{\mu} \Delta$ then $\Delta(\{P' | P' \models F\}) < p$. By the minimal deviation assumption there is an $m' \in \mathbb{N}$ such that $\Delta(\{P' | P' \models F\}) = m' \cdot \pi$, where $m' \leq K - 1$. Therefore,

$$
\begin{aligned}
&\mathcal{P}_{\mu.t_F,P}(1_\mu : E_F) =\\
&\quad = \sum_{P' \in \lceil \Delta \rceil} \Delta(P') \cdot \mathcal{P}_{t_F,P'}(E_F)\\
&\quad = \sum_{\{P' \in \lceil \Delta \rceil \mid P' \models F\}} \Delta(P') \cdot \mathcal{P}_{t_F,P'}(E_F) + \sum_{\{P' \in \lceil \Delta \rceil \mid P' \not\models F\}} \Delta(P') \cdot \mathcal{P}_{t_F,P'}(E_F)\\
&\quad \leq \sum_{\{P' \in \lceil \Delta \rceil \mid P' \models F\}} \Delta(P') \cdot \mathcal{P}_{t_F,P'}(E_F) + \sum_{\{P' \in \lceil \Delta \rceil \mid P' \not\models F\}} \Delta(P') \cdot \delta_F && \text{(by HI)}\\
&\quad \leq (K - 1) \cdot \pi + \delta_F\\
&\quad \overset{def}{=} \gamma'.
\end{aligned}
$$

If $P \not\models F$ and $P \stackrel{\mu}{\not\rightarrow}$ then it holds by Lemma 2.15(1) that $\mathcal{P}_{\mu.t_F,P}(1_\mu : E_F) = 0 \leq (K-1) \cdot \pi + \delta_F$ as well.
Suppose that δ_F satisfies the following property:

$$
(*) \quad \gamma' + \frac{1}{4} \cdot \pi < K - \frac{1}{2} \cdot \pi < \gamma - \frac{1}{4} \cdot \pi.
$$

Consider now the random variable X on the set $O_{\mu.t}$ such that $X(e) = 1$ if $e \in 1_\mu : E_F$ and $X(e) = 0$ if $e \notin 1_\mu : E_F$. We have that:

$$
\begin{aligned}
\mathcal{P}_{\mu.t,P}(X = 1) &= \mathcal{P}_{\mu.t,P}(1_\mu : E_F)\\
\mathcal{P}_{\mu.t,P}(X = 0) &= \mathcal{P}_{\mu.t,P}((1_\mu : E_F)^c) =\\
&= 1 - \mathcal{P}_{\mu.t,P}(1_\mu : E_F) && \text{(by Lemma 2.15(3)).}
\end{aligned}
$$

Therefore, X is a Bernoulli random variable whose parameter is $\mathcal{P}_{\mu.t,P}(1_\mu : E_F)$. The mean of X is $\mathcal{P}_{\mu.t,P}(1_\mu : E_F)$ and its variance is $\mathcal{P}_{\mu.t,P}(1_\mu : E_F) \cdot (1 - \mathcal{P}_{\mu.t,P}(1_\mu : E_F))$. For any $n \in \mathbb{N}$, let X^n be the random variable on the set $O_{(\mu.t)^n}$ such that:

$$X^n(e_1, \dots, e_n) = |\{e_i | 1 \leq i \leq n \text{ and } X(e_i) = 1\}|.$$

The random variable X^n associates to every n-ary sequence of observations in $O_{\mu.t}$ the relative number of observations belonging to $1_\mu : E_F$ and it holds that:

$$X^n(e_1, \dots, e_n) = \sum_{i=1}^{n} X(e_i).$$

Since X is a Bernoulli random variable whose parameter is $\mathcal{P}_{\mu.t,P}(1_\mu : E_F)$, this is equivalent to saying that X^n is a binomial random variable whose parameters are $(n, \mathcal{P}_{\mu.t,P}(1_\mu : E_F))$. Hence, X^n has mean $n \cdot \mathcal{P}_{\mu.t,P}(1_\mu : E_F)$ and its variance is $n \cdot \mathcal{P}_{\mu.t,P}(1_\mu : E_F) \cdot (1 - \mathcal{P}_{\mu.t,P}(1_\mu : E_F))$.

Let $n \in \mathbb{N}$. Define the the following set of observations E relative to the test $(a.t)^n$:

$$E = \{(e_1, \dots, e_n) | \frac{X^n}{n} \geq K - \frac{1}{2} \cdot \pi\},$$

The set E consists of all and only the n-ary sequences of observations in $O_{\mu.t}$ whose number of observations belonging to the set $1_\mu : E_F$ is greater than or equal to $(K - \frac{1}{2} \cdot \pi) \cdot n$. The mean and the variance of the random variable $\frac{X^n}{n}$ respectively are:

$$\begin{aligned} m\left(X^n\right) &= \mathcal{P}_{\mu.t,P}\left(1_\mu : E_F\right) \\ v\left(X^n\right) &= \frac{m\left(X^n\right) \cdot \left(1 - m\left(X^n\right)\right)}{n}. \end{aligned}$$

If $\frac{X^n}{n} < K - \frac{1}{2} \cdot \pi$ then $\mathcal{P}_{\mu.t,P}\left(1_\mu : E_F\right) - \frac{X^n}{n} > \mathcal{P}_{\mu.t,P}\left(1_\mu : E_F\right) - \left(K - \frac{1}{2} \cdot \pi\right)$. By $(*)$ we have that $\mathcal{P}_{\mu.t,P}\left(1_\mu : E_F\right) - \left(K - \frac{1}{2} \cdot \pi\right) \geq \gamma - \left(K - \frac{1}{2} \cdot \pi\right) > \frac{1}{4} \cdot \pi$, therefore $\left|\frac{X^n}{n} - \mathcal{P}_{\mu.t,P}\left(1_\mu : E_F\right)\right| > \mathcal{P}_{\mu.t,P}\left(1_\mu : E_F\right) - \frac{X^n}{n} > \frac{1}{4} \cdot \pi$. So, $\frac{X^n}{n} < K - \frac{1}{2} \cdot \pi$ implies that $|\frac{X^n}{n} - \mathcal{P}_{\mu.t,P}\left(1_\mu : E_F\right)| > \frac{1}{4} \cdot \pi$. As a consequence,

$$(**) \quad \mathcal{P}_{(\mu.t)^n,P}\left(\frac{X^n}{n} < K - \frac{1}{2} \cdot \pi\right) \leq \mathcal{P}_{(\mu.t)^n,P}\left(\left|\frac{X^n}{n} - \mathcal{P}_{\mu.t,P}\left(1_\mu : E_F\right)\right| > \frac{1}{4} \cdot \pi\right).$$

If $\frac{X^n}{n} \geq K - \frac{1}{2} \cdot \pi$ we analogously derive from $(*)$ that:

$$(***) \quad \mathcal{P}_{(\mu.t)^n,P}\left(\frac{X^n}{n} \geq K - \frac{1}{2} \cdot \pi\right) \leq \mathcal{P}_{(\mu.t)^n,P}\left(\left|\frac{X^n}{n} - \mathcal{P}_{\mu.t,P}\left(1_\mu : E_F\right)\right| > \frac{1}{4} \cdot \pi\right).$$

Suppose that $P \models \langle \mu \rangle_p F$. Then it holds that:

$$\begin{aligned}
&\mathcal{P}_{(\mu.t)^n,P}(E) = \mathcal{P}_{(\mu.t)^n,P}\left(\frac{X^n}{n} \geq K - \frac{1}{2}\cdot\pi\right)\\
&= 1 - \mathcal{P}_{(\mu.t)^n,P}\left(\frac{X^n}{n} < K - \frac{1}{2}\cdot\pi\right)\\
&\geq 1 - \mathcal{P}_{(\mu.t)^n,P}\left(\left|\frac{X^n}{n} - \mathcal{P}_{\mu.t,P}(1_\mu : E_F)\right| > \frac{1}{4}\cdot\pi\right) && \text{(By (**))}\\
&= 1 - \mathcal{P}_{(\mu.t)^n,P}\left(\left|\frac{X^n}{n} - m(X^n)\right| > \frac{1}{4}\cdot\pi\right)\\
&\geq 1 - \left(\frac{4}{\pi}\right)^2 \cdot v(X^n) && \text{(By Chebyshev's inequality)}\\
&= 1 - \left(\frac{4}{\pi}\right)^2 \cdot \frac{\mathcal{P}_{\mu.t,P}(1_\mu : E_F)\cdot(1 - \mathcal{P}_{\mu.t,P}(1_\mu : E_F))}{n}\\
&\geq 1 - \left(\frac{4}{\pi}\right)^2 \cdot \frac{1}{n}.
\end{aligned}$$

Conversely, if $P \not\models \langle \mu \rangle_p F$ then:

$$\begin{aligned}
&\mathcal{P}_{(\mu.t)^n,P}(E) = \mathcal{P}_{(\mu.t)^n,P}\left(\frac{X^n}{n} \geq K - \frac{1}{2}\cdot\pi\right)\\
&\leq \mathcal{P}_{(\mu.t)^n,P}\left(\left|\frac{X^n}{n} - \mathcal{P}_{\mu.t,P}(1_\mu : E_F)\right| > \frac{1}{4}\cdot\pi\right) && \text{(By (***))}\\
&\leq \left(\frac{4}{\pi}\right)^2 \cdot v(X^n) && \text{(By Chebyshev's inequality)}\\
&\leq \left(\frac{4}{\pi}\right)^2 \cdot \frac{1}{n}.
\end{aligned}$$

For any level of significance $\delta > 0$, we choose δ_F such that $(*)$ is satisfied and we choose n such that:

$$\delta \geq \left(\frac{4}{\pi}\right)^2 \cdot \frac{1}{n}.$$

We thereby have that the test $(\mu.t)^n$ and the set of observations E satisfy the definition of testability with respect to δ and $\langle \mu \rangle_p F$:

if $P \models \langle \mu \rangle_p F$ then $\mathcal{P}_{(\mu.t)^n,P}(E) \geq 1 - \left(\frac{4}{\pi}\right)^2 \cdot \frac{1}{n} \geq 1 - \delta$,

if $P \not\models \langle \mu \rangle_p F$ then $\mathcal{P}_{(\mu.t)^n,P}(E) \leq \left(\frac{4}{\pi}\right)^2 \cdot \frac{1}{n} \leq \delta$.

□

2.3.2 Characterization of probabilistic bisimilarity

The ls-testability of Probabilistic Modal Logic formulas allows us to prove that in Larsen and Skou's testing scenario two processes are probabilistically bisimilar if and only if they have the same probability of doing the same observations when the a test is executed (Theorem 2.18).

Lemma 2.17. *Let $\mathcal{T} = \langle St, \mathscr{A}, \longrightarrow \rangle$ be a reactive pLTS satisfying the minimal deviation assumption. If $P \sim Q$ then $\mathcal{P}_{t,P}(e) = \mathcal{P}_{t,Q}(e)$ for all tests t in $\mathbf{T}^{\mathsf{LS}}$ and for all observations $e \in O_t$.*

Proof. The proof is by induction on the structure of the tests in $\mathbf{T}^{\mathsf{LS}}$. Suppose that $P \sim Q$.

$(t = \omega)$ There is only one possible observation (i.e. the observation $e = 1_\omega$) and $\mathcal{P}_{t,P}(e) = \mathcal{P}_{t,Q}(e) = 1$.

$(t = \mu.t')$ There are two possibilities:

$e = 0_\mu$ If $P \xrightarrow{\mu}$ then it follows from $P \sim Q$ that $Q \overset{\mu}{\not\longrightarrow}$ and $\mathcal{P}_{t,P}(e) = \mathcal{P}_{t,Q}(e) = 0$. If $P \overset{\mu}{\not\longrightarrow}$ then $Q \overset{\mu}{\not\longrightarrow}$ and $\mathcal{P}_{t,P}(e) = \mathcal{P}_{t,Q}(e) = 1$.

$e = 1_\mu : e'$ If $P \overset{\mu}{\not\longrightarrow}$ then both the probability values are equal to 0, analogously to the previous case. Now, suppose that $P \xrightarrow{\mu} \Delta$. By the hypothesis that $P \sim Q$, there is a Θ such that $Q \xrightarrow{\mu} \Theta$ and $\Delta \overline{\sim} \Theta$. Let I be a finite index set through which we can decompose Δ and Θ, according to Definition 2.3. We have that:

$$\begin{aligned}
\mathcal{P}_{\mu.t',P}(1_\mu : e') &= \sum_{P' \in \lceil \Delta \rceil} \Delta(P') \cdot \mathcal{P}_{t',P'}(e') \\
&= \sum_{P' \in \lceil \Delta \rceil} \left(\sum_{\{i \in I | P_i = P'\}} p_i \right) \cdot \mathcal{P}_{t',P'}(e') \\
&= \sum_{P' \in \lceil \Delta \rceil} \sum_{\{i \in I | P_i = P'\}} p_i \cdot \mathcal{P}_{t',P_i}(e') \\
&= \sum_{i \in I} p_i \cdot \mathcal{P}_{t',P_i}(e') \\
&= \sum_{i \in I} p_i \cdot \mathcal{P}_{t',Q_i}(e') && \text{(by HI)} \\
&= \sum_{Q' \in \lceil \Theta \rceil} \Theta(Q') \cdot \mathcal{P}_{t',Q'}(e') \\
&= \mathcal{P}_{\mu.t',Q}(1_\mu : e').
\end{aligned}$$

$(t = (t_1, ..., t_n))$ The result directly follows from the inductive hypothesis.

□

Theorem 2.18 (Larsen & Skou's Theorem)**.** *Let $\mathcal{T} = \langle St, \mathscr{A}, \longrightarrow\rangle$ be a reactive pLTS satisfying the minimal deviation assumption. Then $P \sim Q$ if and only if $\mathcal{P}_{t,P}(e) = \mathcal{P}_{t,Q}(e)$ for all tests t in $\mathbf{T}^{\mathsf{LS}}$ and for all observations $e \in O_t$.*

Proof. We proved the left to right implication in Lemma 2.17.
As for the right to left implication, suppose that $P \not\sim Q$. By Theorem 2.12, there is a PML-formula F such that $P \models F$ and $Q \not\models F$. Let δ be a positive real number. By Theorem 2.16, there is a test t_δ in $\mathbf{T}^{\mathsf{LS}}$ and there is a set of observation E_δ such that:

$$(*) \quad \mathcal{P}_{t_\delta,P}(E_\delta) \geq 1-\delta \quad \text{and} \quad \mathcal{P}_{t_\delta,Q}(E_\delta) \leq \delta.$$

If $\mathcal{P}_{t,P}(e) = \mathcal{P}_{t,Q}(e)$ for all tests t in $\mathbf{T}^{\mathsf{LS}}$ and for all observations $e \in O_t$, then $\mathcal{P}_{t_\delta,P}(E_\delta) = \mathcal{P}_{t_\delta,Q}(E_\delta)$, which contradicts $(*)$. Then there is a test t_δ in $\mathbf{T}^{\mathsf{LS}}$ and there is an observation $e_\delta \in O_t$ such that $\mathcal{P}_{t_\delta,P}(E_\delta) \neq \mathcal{P}_{t_\delta,Q}(E_\delta)$, and the result follows by contraposition.

□

Chapter 3

Testing processes through higher-order languages

This chapter introduces three testing-based equivalences on processes: *may*-equivalence, *must*-equivalence and *test*-equivalence. A higher-order concurrent language named HOL is defined in Section 3.2 and two testing scenarios based on this language are analyzed. In the first one (Section 3.3), the tested processes are nondeterministic processes and we prove that ready simulation equivalence coincides with both *may*-equivalence and *test*-equivalence. In the second testing scenario (Section 3.4), the tested processes are probabilistic reactive processes and the three testing equivalences collapse onto probabilistic bisimilarity.

3.1 Testing equivalences

Testing equivalences on processes are based on the idea that, given a class of tests, two processes P and Q are equivalent if and only if there is no test distinguishing P and Q.
The testing approach to behavioral equivalences was developed by Rocco De Nicola and Matthew Hennessy (De Nicola and Hennessy 1984) in the setting of nondeterministic systems. They proved that if we use CCS-processes to test CCS-processes, the induced testing equivalence is the so-called "failure equivalence", which is finer than completed trace equivalence but coarser than ready simulation equivalence (van Glabbeek 2001). In (Abramsky 1987), Samson Abramsky proposed a stronger language of tests and proved that bisimilarity coincides with the equivalence induced by these tests on Labelled Transition Systems. In (Yi and Larsen 1992), Wang Yi and Kim G. Larsen extended De Nicola and Hennessy's testing theory to the setting of probabilistic processes.
In order to define testing-based equivalences on processes, we need to set up a testing scenario.

Definition 3.1. A *testing scenario* is a quadruple $\langle \mathbf{P}, \mathbf{T}, \mathcal{O}, \mathit{Succ} \rangle$, where:

- $\mathbf{P}$ is a set of processes,
- $\mathbf{T}$ is a set of tests which can be applied to processes,
- $\mathcal{O}$ is a set of values (representing the possible outcomes from applying a test to a process) equipped with a partial order $\leq$,
- $\mathit{Succ} : \mathbf{T} \times \mathbf{P} \to \mathscr{P}^{+}(\mathcal{O})$ is a function such that $\mathit{Succ}(T, P)$ is the non-empty set of the possible results of applying the test T to the process P.

We define the associated *may*-preorder, *must*-preorder and *test*-preorder on $\mathbf{P}$ as follows:

$$\begin{aligned} P \sqsubseteq^{\mathbf{T}}_{\text{may}} Q &\text{ iff } \quad \forall T \in \mathbf{T}, \bigsqcup \mathit{Succ}(T,P) \leq \bigsqcup \mathit{Succ}(T,Q) \\ P \sqsubseteq^{\mathbf{T}}_{\text{must}} Q &\text{ iff } \quad \forall T \in \mathbf{T}, \bigsqcap \mathit{Succ}(T,P) \leq \bigsqcap \mathit{Succ}(T,Q) \\ P \sqsubseteq^{\mathbf{T}}_{\text{test}} Q &\text{ iff } \quad P \sqsubseteq^{\mathbf{T}}_{\text{may}} Q \text{ and } P \sqsubseteq^{\mathbf{T}}_{\text{must}} Q \end{aligned}$$

The induced equivalences are denoted by $\simeq^{\mathbf{T}}_{\text{may}}$, $\simeq^{\mathbf{T}}_{\text{must}}$ and $\simeq^{\mathbf{T}}_{\text{test}}$, respectively.

The interpretation of the testing preorders and of the related equivalences will become clear in the following sections, where we discuss two specific testing scenarios based on the language HOL.

3.2 HOL

We define the higher-order language HOL, freely inspired by the Kell Calculus of (Schmitt and Stefani 2005). In a higher-order language we have variables ranging over the terms of the language and we have constructs allowing terms to take other terms as input. HOL is a process calculus featuring some of the CCS operators that we introduced in Section 1.4: the **0** operator, the prefix operator and the parallel operator. Besides these operators, the HOL language features constructs for kells[37], passivation of kells and refusal of actions. In this section we only define the syntax of HOL, leaving the definition of its semantics to the following sections.
Let us briefly list and describe the main facilities supplied by this language.

Hierarchical localities with names A process can be in a kell with a certain name, and this locality can contain sublocalities with possibly different names as well. For instance, $P \mid \langle Q \rangle_h$ is the parallel composition of processes whose right-side component is the

[37] In the Kell Calculus, kells are localities identified by a name. As Alan Schmitt and Jean-Bernard Stefani put it: "the word 'kell' is intended to remind of the word 'cell', in a loose analogy with biological cells"(Schmitt and Stefani 2005: 149).

process Q contained in the locality named h. At the same time, this whole process could be placed into a higher-level locality named l, as in $\langle P \,|\, \langle Q\rangle_h\rangle_l$. The processes in the environment of $\langle P \,|\, \langle Q\rangle_h\rangle_l$ (i.e. in parallel composition with $\langle P \,|\, \langle Q\rangle_h\rangle_l$) can communicate with P, while only P can interact with $\langle Q\rangle_h$.

Local communication Besides the usual CCS prefix operator, the HOL language has local prefixes, i.e. prefixes where the action is indexed with the name of a kell. For instance, the process $a^{\rangle l}.P$ can lead to a synchronization whenever it is in parallel composition with a process performing an $\bar{a}$-labelled transition inside a kell named l. Instead, $a.P$ can only interact with processes on its own level, that is, it cannot communicate with any process inside a kell.

Local refusal We use $\widetilde{a}^l.P$ to denote the process which tests the availability in the environment of a locality named l where it is not possible to perform an a-labelled transition. If there is a locality satisfying this condition in the environment of $\widetilde{a}^l.P$, then the locality is destroyed while P is enabled.

Passivation of kells The higher-order construct of the HOL language is the passivation operator $\mathtt{pass}(x)_l$, where x is a variable ranging over processes and l is the name of a locality. A kell $\langle Q\rangle_l$ is passivated by $\mathtt{pass}(x)_l.P$, which takes as input Q and substitutes it to the free occurrences of the higher-order variable x in P.

Definition 3.2. Let $\mathscr{N}$ be a set of atomic actions $a, b, c....$

- $\overline{\mathscr{N}}$ is the set of its conames, i.e. the set $\{\bar{a} |\, a \in \mathscr{N}\}$ where $a = \bar{\bar{a}}$
- $\mathscr{A}^\star = \mathscr{N} \cup \overline{\mathscr{N}} \cup \{\alpha^{\rangle l} |\, \alpha \in \mathscr{N} \cup \overline{\mathscr{N}}, l \in \mathscr{N}\}$
- $\mathscr{A}^*_{\tau,\omega} = \mathscr{A}^\star \cup \{\tau, \omega\}$, where ω and τ are two names of actions that do not appear in $\mathscr{A}^\star$,
- $\widetilde{\mathscr{A}^\star} = \{\widetilde{\alpha}^{\rangle l} | \alpha \in \mathscr{N} \cup \overline{\mathscr{N}}, l \in \mathscr{N}\} \cup \{\overline{\widetilde{\alpha}}^{\rangle l} |\, \alpha \in \mathscr{N} \cup \overline{\mathscr{N}}, l \in \mathscr{N}\}$

We let $\alpha, \beta...$ range over $\mathscr{N} \cup \overline{\mathscr{N}}$ and $\alpha^*, \beta^*...$ range over $\mathscr{A}^*$. Define $\bar{\mu} = \bar{\alpha}^{\rangle l}$ if $\mu = \alpha^{\rangle l}$ and $\bar{\mu} = \bar{\widetilde{\alpha}}^{\rangle l}$ if $\mu = \widetilde{\alpha}^{\rangle l}$.
The HOL-terms on $\mathscr{N}$ are defined by the following grammar:

$$H ::= \mathbf{0} \;\Big|\; x \;\Big|\; \alpha^*.H \;\Big|\; \widetilde{\alpha}^{\rangle l}.H \;\Big|\; \omega.H \;\Big|\; H \,|\, H \;\Big|\; \langle H\rangle_l \;\Big|\; \mathtt{pass}(x)_l.H$$

where x is a variable in a countably infinite set of variables $\mathcal{V}$.

In the remaining part of this work we will constantly use the contexts of the language HOL and apply them to terms that are not already embodied in the language defined above. Given a language, a context of the language is a term where one or more empty holes appear. Formally, the HOL-contexts on a set of actions' names $\mathscr{N}$ are defined by the following grammar:

$$C ::= \mathbf{0} \;\Big|\; x \;\Big|\; [\cdot] \;\Big|\; \alpha^{*}.C \;\Big|\; \widetilde{\alpha}^{\rangle l}.C \;\Big|\; \omega.C \;\Big|\; C\,|\,C \;\Big|\; \langle C\rangle_l \;\Big|\; \mathtt{pass}(x)_l.C$$

where $[\cdot]$ represents an empty hole.
We let $Ctx(\mathsf{HOL})$ denote the set of HOL-contexts on a given set of actions' names $\mathscr{N}$.

3.3 Testing nondeterministic processes

In order to apply HOL-contexts to nondeterministic processes, we define a new language HOLn. This language is obtained by adding to the syntax of HOL infinitely many constant's names, one for every nondeterministic process.

Definition 3.3. The terms of the language HOLn on $\mathscr{N}$ are defined by the following grammar:

$$P ::= \mathbf{0} \;\Big|\; x \;\Big|\; S \;\Big|\; \alpha^{*}.P \;\Big|\; \widetilde{\alpha}^{\rangle l}.P \;\Big|\; \omega.P \;\Big|\; P\,|\,P \;\Big|\; \langle P\rangle_l \;\Big|\; \mathtt{pass}(x)_l.P$$

where S is (the name of) a nondeterministic process whose transitions are labelled with actions in $\mathscr{N}$.
The set $Pr(\mathsf{HOLn})$ of HOLn's processes is the set of HOLn's closed terms.

Let C be a HOLn-context where n distinct holes appear and let $P_1, ..., P_n$ be HOLn-terms. We say that C is an n-ary HOLn-context and we write $C[P_1, ..., P_n]$ to denote the HOLn-term obtained by substituting P_i to the i-th occurrence of $[\cdot]$ in C, for $1 \leq i \leq n$.
The operational semantic of the language HOLn on $\mathscr{N}$ is defined by the LTS

$$\langle Pr(\mathsf{HOLn}), \mathscr{A}^{\mathsf{HOL}}, \longrightarrow\rangle,$$

where $\mathscr{A}^{\mathsf{HOL}} = \mathscr{A}^{*}_{\tau,\omega} \cup \widetilde{\mathscr{A}^{*}} \cup \{\mathtt{pass}(\rho)_l |\ \rho \in Pr(\mathsf{HOLn}) \cup \mathcal{V} \text{ and } l \in \mathscr{N}\}$ and the transition relation is given by the rules in figure 3.1. A process P is stable (see rule (Oref)) if it cannot perform τ-labelled transitions.
Henceforth, we will use the following notation:

- we write μ instead of $\mu.\mathbf{0}$, where $\mu \in \mathscr{A}^{\star} \cup \widetilde{\mathscr{A}^{\star}}$,
- for any $n \geq 1$, we inductively define the parallel composition $\mathop{\Big|\Big|}_{i=1}^{n} P_i$ of the HOLn-terms $P_1, ..., P_n$ as follows:

$$\mathop{\Big|\Big|}_{i=1}^{1} P_i = P_1 \qquad\qquad \mathop{\Big|\Big|}_{i=1}^{n+1} P_i = (\mathop{\Big|\Big|}_{i=1}^{n} P_i)\,|\,P_{n+1}$$

$$\frac{S \xrightarrow{a}{}' S'}{S \xrightarrow{a} S'}\,(NP)$$ where $\longrightarrow'$ is the transition relation of the LTS for S

$$\frac{}{\mu.P \xrightarrow{\mu} P}\,(\text{pref})_{\mu \in \mathscr{A}^\star \cup \{\omega\}}$$

$$\frac{}{\widetilde{\alpha}^{\rangle l}.P \xrightarrow{\widetilde{\alpha}\rangle l} P}\,(\text{Iref})$$

$$\frac{P \overset{\alpha}{\not\longrightarrow} \text{ and } P \text{ is stable}}{\langle P\rangle_l \xrightarrow{\overline{\widetilde{\alpha}}\rangle l} \mathbf{0}}\,(\text{Oref})$$

$$\frac{P_1 \xrightarrow{\mu} P_1'}{P_1 \,|\, P_2 \xrightarrow{\mu} P_1' \,|\, P_2}\,(\text{parL})_{\mu \in \mathscr{A}^{\mathsf{HOLn}}}$$

$$\frac{P_1 \xrightarrow{\mu} P_1'}{P_2 \,|\, P_1 \xrightarrow{\mu} P_2 \,|\, P_1'}\,(\text{parR})_{\mu \in \mathscr{A}^{\mathsf{HOLn}}}$$

$$\frac{P_1 \xrightarrow{\mu} P_1' \qquad P_2 \xrightarrow{\bar{\mu}} P_2'}{P_1 \,|\, P_2 \xrightarrow{\tau} P_1' \,|\, P_2'}\,(\text{sync})_{\mu \in \mathscr{A}^\star \cup \widetilde{\mathscr{A}}^\star}$$

$$\frac{P \xrightarrow{\alpha} P'}{\langle P\rangle_l \xrightarrow{\alpha\rangle l} \langle P'\rangle_l}\,(\text{kell})$$

$$\frac{P \xrightarrow{\tau} P'}{\langle P\rangle_l \xrightarrow{\tau} \langle P'\rangle_l}\,(\tau\text{kell})$$

$$\frac{}{\mathtt{pass}(x)_l.P \xrightarrow{\mathtt{pass}(x)_l} P}\,(\text{Ipass})$$

$$\frac{}{\langle P\rangle_l \xrightarrow{\mathtt{pass}(P)_l} \mathbf{0}}\,(\text{Opass})$$

$$\frac{P_1 \xrightarrow{\mathtt{pass}(\rho_1)_l} P_1' \qquad P_2 \xrightarrow{\mathtt{pass}(\rho_2)_l} P_2'}{P_1 \,|\, P_2 \xrightarrow{\tau} P_1' \,|\, P_2'\{^P/_x\}}\,(\text{psync})$$ where either $\rho_1 = P$ and $\rho_2 = x$ or $\rho_1 = x$ and $\rho_2 = P$

Figure 3.1: SOS for processes in HOLn.

3.3.1 Testing preorders on nondeterministic processes

We use HOLn-contexts as tests for nondeterministic processes. There are only two possible outcomes: success, represented by $\top$, and failure, represented by $\bot$. The test is executed by substituting a process to the empty hole of the HOL-context playing the role of the test. The function $\mathbb{S}$ associates to a HOL-process P a non-empty subset of $\{\bot, \top\}$, determined as follows:

$$\begin{aligned}\mathbb{S}(P) =& \{\top \mid P \Longrightarrow S \text{ for some } S \text{ such that } S \xrightarrow{\omega}\}\\ &\cup \{\bot \mid P \Longrightarrow S \text{ for some } S \text{ such that } S \not\xrightarrow{\omega} \text{ and } S \not\xrightarrow{\tau}\}\\ &\cup \{\bot \mid P \text{ diverges, i.e. there is an infinite } \tau\text{-labelled path from } P\}.\end{aligned}$$

Hence, in this testing scenario the function *Succ* is such that $\mathit{Succ}(C, P) = \mathbb{S}(C[P]) \subseteq \{\bot, \top\}$, for every unary HOL-context C and for every nondeterministic process P.
Define the following relations between processes and unary HOL-contexts:

P **may pass** C if and only if there exists an S such that $C[P] \Longrightarrow S \xrightarrow{\omega}$,

P **must pass** C if and only if for every τ-labelled path $C[P] = S_0 \xrightarrow{\tau} S_1 \xrightarrow{\tau} S_2 \ldots$ there is an $n \in \mathbb{N}$ such that $S_n \xrightarrow{\omega}$.

By the definition of the function *Succ*, in the testing scenario $\langle Pr, Ctx(\mathsf{HOL}), \{\bot, \top\}, Succ\rangle$, where Pr is the set of nondeterministic processes whose transitions are labelled with actions in $\mathscr{N}$, the testing preorders introduced in Section 3.1 can be characterized as follows:

$$\begin{aligned}P \sqsubseteq^{\mathsf{HOLn}}_{\text{may}} Q &\text{ iff } \top \in \mathbb{S}(C[P]) \text{ implies } \top \in \mathbb{S}(C[Q]) \text{ for every unary } \mathsf{HOLn}\text{-context } C\\ &\text{ iff } P \textbf{ may pass } C \text{ implies } Q \textbf{ may pass } C \text{ for every unary } \mathsf{HOLn}\text{-context } C\end{aligned}$$

$$\begin{aligned}P \sqsubseteq^{\mathsf{HOLn}}_{\text{must}} Q &\text{ iff } \{\top\} = \mathbb{S}(C[P]) \text{ implies } \{\top\} = \mathbb{S}(C[Q]) \text{ for every unary } \mathsf{HOLn}\text{-context } C\\ &\text{ iff } P \textbf{ must pass } C \text{ implies } Q \textbf{ must pass } C \text{ for every unary } \mathsf{HOLn}\text{-context } C\end{aligned}$$

$$P \sqsubseteq^{\mathsf{HOLn}}_{\text{test}} Q \text{ iff } \mathbb{S}(C[P]) \subseteq \mathbb{S}(C[Q]).$$

3.3.2 Ready simulation equivalence implies *test*-equivalence

This section is devoted to proving that if two nondeterministic processes P and Q are ready simulation equivalent then they are both *may*-equivalent and *must*-equivalent in the testing scenario defined above.
By Theorem 3.4, if P and Q are ready simulation equivalent processes whose transitions are labelled with actions in $\mathscr{N}$ then the processes $C[P]$ and $C[Q]$ are themselves ready simulation equivalent with respect to a set of actions including τ and ω, for any unary HOL-context C. As a consequence, proving that ready simulation equivalent nondeterministic processes whose transitions are labelled with action in $\mathscr{N}$ are *may*-equivalent (respectively: *must*-equivalent) boils down to proving that if P and Q are ready simulation equivalent nondeterministic processes defined over a set of atomic actions including τ and ω then P has a successful path (respectively: all the paths from P are successful) if and only if

Q has a successful path (respectively: all the paths from Q are successful). This result is essential for the proofs of Theorem 3.5 and Theorem 3.7, whose immediate corollaries are that ready simulation equivalence implies *may*-equivalence and that ready simulation equivalence implies *must*-equivalence, respectively.

Theorem 3.4. *Let $\mathcal{T} = \langle St, \mathscr{N}, \longrightarrow\rangle$ be an LTS. For any $n \in \mathbb{N}$, let $P_1, ..., P_n, Q_1, ..., Q_n$ be processes in $\mathcal{T}$ such that $P_i \precsim^r Q_i$, for $1 \le i \le n$. For every n-ary* HOL*-context C, $C[P_1, ..., P_n] \precsim^r C[Q_1, ..., Q_n]$, where $\precsim^r$ is defined on $\mathscr{A}^*_{\tau,\omega}$.*

Proof. Consider the following relation on HOLn-processes:

$$\begin{aligned}\mathcal{R} = \{< C[P_1, ..., P_n], C[Q_1, ..., Q_n] > \mid n \in \mathbb{N},\ & C \text{ is an } n\text{-ary } \mathsf{HOL}\text{-context},\\ & \{P_1, ..., P_n, Q_1, ..., Q_n\} \subseteq St \text{ and } P_i \lesssim^r Q_i \text{ for } 1 \le i \le n\}.\end{aligned}$$

We show that $\mathcal{R}$ is a ready simulation, that is, we prove by structural induction on C that for every $n \in \mathbb{N}$, whenever C is an n-ary HOL-context and $C[P_1, ..., P_n] \,\mathcal{R}\, C[Q_1, ..., Q_n]$,

1. for every $\mu \in \mathscr{A}^*_{\tau,\omega}$, if $C[P_1, ..., P_n] \xrightarrow{\mu} S$ then $C[Q_1, ..., Q_n] \xrightarrow{\mu} S'$ and $S \,\mathcal{R}\, S'$, for some $S' \in$ HOLn,

2. for every $\mu \in \mathscr{A}^*_{\tau,\omega}$, if $C[P_1, ..., P_n] \stackrel{\mu}{\not\longrightarrow}$ then $C[Q_1, ..., Q_n] \stackrel{\mu}{\not\longrightarrow}$.

($C = [\cdot]$) C is a unary context and $C[P] = P \lesssim^r Q = C[Q]$ satisfies the conditions by the definition of $\lesssim^r$.

($C = \omega.C'$), ($C = \alpha^*.C'$) The result directly follows from the inductive hypothesis on C'.

($C = \widetilde{\alpha}^*.C'$), ($C = \mathtt{pass}(x)_l.C'$) The processes cannot perform transitions labelled with actions in $\mathscr{A}^*_{\tau,\omega}$, hence they vacuously satisfy the first condition.

($C = \langle C'\rangle_l$) $\langle C'\rangle_l[P_1, ..., P_n] = \langle C'[P_1, ..., P_n]\rangle_l$ and there are two cases:

- $C'[P_1, ..., P_n] \xrightarrow{\alpha} S$, so by rule (kell) $\langle C'[P_1, ..., P_n]\rangle_l \xrightarrow{\alpha^l} \langle S\rangle_l$.
  It follows from the inductive hypothesis that $\langle C'[Q_1, ..., Q_n]\rangle_l \xrightarrow{\alpha^l} \langle S'\rangle_l$, for some S' such that $S \,\mathcal{R}\, S'$. By the definition of $\mathcal{R}$, we have that that there is an $m \in \mathbb{N}$, there is an m-ary HOLn-context C'' and there are processes $P'_1, ..., P'_m, Q'_1, ..., Q'_m$ in $\mathcal{T}$ such that $S = C''[P'_1, ..., P'_m]$, $S' = C''[Q'_1, ..., Q'_m]$ and $P'_i \,\mathcal{R}\, Q'_i$, for all i from 1 to m. Therefore, there is an m-ary HOLn-context $C''' = \langle C''\rangle_l$ such that $\langle S\rangle_l = C'''[P'_1, ..., P'_m] \,\mathcal{R}\, C'''[Q'_1, ..., Q'_m] = \langle S'\rangle_l$.
- $C'[P_1, ..., P_n] \xrightarrow{\tau} S$ and $\langle C'[P_1, ..., P_n]\rangle_l \xrightarrow{\tau} \langle S\rangle_l$, by rule ($\tau$kell).
  The proof of the first condition is analogous to the previous case.

We now prove that the second condition holds as well. If $\langle C'[Q_1, ..., Q_n]\rangle_l \xrightarrow{\mu}$ then $\mu \in \{\alpha^*, \tau\}$ and $C'[Q_1, ..., Q_n]$ can perform an α-labelled, or τ-labelled, action. By the inductive hypothesis, $C'[P_1, ..., P_n]$ can perform a transition labelled with the same name, so $\langle C'[Q_1, ..., Q_n]\rangle_l \xrightarrow{\mu}$.

($C = C_1 \mid C_2$) $C[P_1, ..., P_n] = C_1[P_1, ..., P_k] \mid C_2[P_{k+1}, ..., P_n]$ and there are five possible cases:

- $C_1[P_1, ..., P_k] \xrightarrow{\mu} S$ and we derive that $C[P_1, ..., P_n] \xrightarrow{\mu} S \mid C_2[P_{k+1}, ..., P_n]$, by rule (parL).
  By the inductive hypothesis there is an S' such that $C_1[Q_1, ..., Q_k] \xrightarrow{\mu} S'$ and $S \mathcal{R} S'$; by rule (parL), $C_1[Q_1, ..., Q_k] \mid C_2[Q_{k+1}, ..., Q_n] \xrightarrow{\mu} S' \mid C_2[Q_{k+1}, ..., Q_n]$. The first condition follows analogously to the case ($C = \langle C'\rangle_l$).
- $C_2[P_{k+1}, ..., P_n] \xrightarrow{\mu} S$ and, by rule (parR), $C[P_1, ..., P_n] \xrightarrow{\mu} C_1[P_1, ..., P_k] \mid S$. Symmetrical to the previous case.
- $C_1[P_1, ..., P_k] \xrightarrow{\alpha^*} S_1$, $C_2[P_{k+1}, ..., P_n] \xrightarrow{\bar{\alpha}^*} S_2$ and $C[P_1, ..., P_n] \xrightarrow{\tau} S_1 \mid S_2$.
  It follows from the inductive hypothesis that by applying rule (sync) we derive that $C_1[Q_1, ..., Q_k] \mid C_2[Q_{k+1}, ..., Q_n] \xrightarrow{\tau} S_1' \mid S_2'$, for some S_1', S_2' such that $S_1 \mathcal{R} S_1'$ and $S_2 \mathcal{R} S_2'$. By the definition of $\mathcal{R}$,

$$\begin{aligned}
&S_1 \mid S_2 = \\
&= C_1'[P_1', ..., P_{m_1}'] \mid C_2'[P_1'', ..., P_{m_2}''] = \\
&= C_1' \mid C_2'[P_1', ..., P_{m_1}', P_1'', ..., P_{m_2}''] \qquad \mathcal{R} \qquad C_1' \mid C_2'[Q_1', ..., Q_{m_1}', Q_1'', ..., Q_{m_2}''] = \\
&\qquad\qquad\qquad\qquad\qquad\qquad\qquad = C_1'[Q_1', ..., Q_{m_1}'] \mid C_2'[Q_1'', ..., Q_{m_2}''] = \\
&\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad = S_1' \mid S_2'.
\end{aligned}$$

  where $P_i' \mathcal{R} Q_i'$ and $P_j'' \mathcal{R} Q_j''$ for $1 \le i \le m_1$ and $1 \le j \le m_2$.
- $C_1[P_1, ..., P_k] \xrightarrow{\mathtt{pass}(x)_l} S_1$, $C_2[P_{k+1}, ..., P_n] \xrightarrow{\mathtt{pass}(S)_l} S_2$ and by rule (psync) we derive that $C[P_1, ..., P_n] \xrightarrow{\tau} S_1 \mid S_2\{S/x\}$.
  $P_1, ..., P_n$ cannot perform higher-order actions, hence $C_1[P_1, ..., P_k] \xrightarrow{\mathtt{pass}(x)_l}$ implies that C_1 is the parallel composition of contexts such that at least one of them is of the form $\mathtt{pass}(x)_l.C_1'$. Analogously, if $C_2[P_{k+1}, ..., P_n] \xrightarrow{\mathtt{pass}(S)_l}$ then C_2 is the parallel composition of contexts such that at least one of them is of the form $\langle C_2'\rangle$. For the sake of simplicity, suppose that $C_1[P_1, ..., P_k] = \mathtt{pass}(x)_l.C_1'[P_1, ..., P_k]$ and $C_2[P_{k+1}, ..., P_n] = \langle C_2'[P_{k+1}, ..., P_n]\rangle_l$. Therefore, $C[P_1, ..., P_n]$ performs a τ-labelled transition to $C_1'[P_1, ..., P_k]\{C_2'[P_{k+1}, ..., P_n]/x\} \mid \mathbf{0}$. The variable x does not appear in $P_1, ..., P_n$, so $C_1'[P_1, ..., P_k]\{C_2'[P_{k+1}, ..., P_n]/x\} = C_1'\{C_2'[P_{k+1}, ..., P_n]/x\}[P_1, ..., P_k]$. Suppose now that the variable x occurs free h times in C_1' and let $e = h \cdot (n-k) +$

k. Thus, we have that $C_1'\{{}^{C_2'[P_{k+1},...,P_n]}/_{x}\} \,|\, \mathbf{0}[P_1, ..., P_k] = C_1'\{{}^{C_2'}/_{x}\} \,|\, \mathbf{0}[P_1', ..., P_e']$, where $P_1', ..., P_e'$ is a sequence of processes in $\{P_1, ..., P_n\}$ preserving the previous substitutions in the e-ary context $C_1'\{{}^{C_2'}/_{x}\}$.
Symmetrically, $C[Q_1, ..., Q_n] \xrightarrow{\tau} C_1'\{{}^{C_2'[Q_{k+1},...,Q_n]}/_{x}\}[Q_1, ..., Q_k] \,|\, \mathbf{0}$, which we can rewrite as $C_1'\{{}^{C_2'}/_{x}\} \,|\, \mathbf{0}[Q_1', ..., Q_e']$, where $P_i' \,\mathcal{R}\, Q_i'$ for all i such that $1 \leq i \leq e$. Therefore, $C_1'\{{}^{C_2'}/_{x}\} \,|\, \mathbf{0}[P_1', ..., P_e'] \,\mathcal{R}\, C_1'\{{}^{C_2'}/_{x}\} \,|\, \mathbf{0}[Q_1', ..., Q_e']$.

- $C_1[P_1, ..., P_k] \xrightarrow{\widetilde{\alpha}\rangle l} S_1$, $C_2[P_{k+1}, ..., P_n] \xrightarrow{\overline{\widetilde{\alpha}}\rangle l} S_2$ and, by rule (sinc), $C[P_1, ..., P_n] \xrightarrow{\tau} S_1 \,|\, S_2$.
  The proof is similar to the previous case: $C_1[P_1, ..., P_k] \xrightarrow{\widetilde{\alpha}\rangle l}$ implies that C_1 is the parallel composition of contexts such that at least one of them is of the form $\widetilde{a}\rangle l.C_1'$, while C_2 must be a parallel composition with a context of the form $\langle C_2' \rangle$ as a component.

We have showed that the first condition of the definition of ready simulation holds if $C = C_1 \,|\, C_2$. Suppose now that $C[Q_1, ..., Q_n] \xrightarrow{\mu}$. If $C_1[Q_1, ..., Q_k] \xrightarrow{\mu}$ (respectively: $C_2[Q_{k+1}, ..., Q_n] \xrightarrow{\mu}$) then by the hypothesis of induction we have that $C_1[P_1, ..., P_k] \xrightarrow{\mu}$ (respectively: $C_2[P_{k+1}, ..., P_n] \xrightarrow{\mu}$), which in turn implies that the process $C[P_1, ..., P_n]$ can perform a μ-labelled transition. If $C[Q_1, ..., Q_n] \xrightarrow{\tau}$ and the transition is derived from a synchronization, the complementary actions performed by the two components of the parallel composition must be performed by $C_1[P_1, ..., P_k]$ and $C_2[P_{k+1}, ..., P_n]$ as well, as a consequence of the inductive hypothesis. Therefore, $C[Q_1, ..., Q_n] \xrightarrow{\mu}$ implies that $C[P_1, ..., P_n] \xrightarrow{\mu}$, which means that the second condition of the definition of ready simulation holds as well.

□

Theorem 3.5. *Let P, Q be processes in an image-finite LTS $\mathcal{T} = \langle St, \mathscr{N}, \longrightarrow \rangle$. If $P \precsim^r Q$ then $P \sqsubseteq^{\mathsf{HOLn}}_{\text{may}} Q$.*

Proof. Let C be a unary HOLn-context. If P **may pass** C then there is an S such that $C[P] \Longrightarrow S \xrightarrow{\omega}$, which in turn implies that for some $n \in \mathbb{N}$,

$$\sigma = \underbrace{\tau \ldots \tau}_{n-times} \omega \in Tr(C[P]).$$

By Theorem 3.4, $P \precsim^r Q$ implies that $C[P] \precsim^r C[Q]$, where $\precsim^r$ is defined over the set of actions $\mathscr{A}^*_{\tau,\omega}$. It follows from Theorem 1.36, Theorem 1.31 and Theorem 1.18 that $C[P]$ and $C[Q]$ are trace equivalent with respect to the set of actions $\mathscr{A}^*_{\tau,\omega}$, so $\sigma \in Tr(C[Q])$. Therefore, Q **may pass** C and we conclude that $P \sqsubseteq^{\mathsf{HOLn}}_{\text{may}} Q$.

□

Lemma 3.6. *Let P, Q be HOLn-processes such that $P \mathrel{\mathcal{R}} Q$, for some ready simulation $\mathcal{R}$ defined over the set of actions $\mathscr{A}^*_{\tau,\omega}$. For all $n \in \mathbb{N}$, if $P \mathrel{\mathcal{R}} Q$ and $P = S_0 \xrightarrow{\tau} S_1 \xrightarrow{\tau} S_2 \ldots \xrightarrow{\tau} S_n$ and $S_i \stackrel{\omega}{\not\longrightarrow}$ for $0 \leq i \leq n$ then there are $S'_0, \ldots, S'_n$ such that $Q = S'_0 \xrightarrow{\tau} S'_1 \xrightarrow{\tau} S'_2 \ldots \xrightarrow{\tau} S'_n$, $S'_i \stackrel{\omega}{\not\longrightarrow}$ and $S_i \mathrel{\mathcal{R}} S'_i$ for $0 \leq i \leq n$.*

Proof. By induction on n.

$(n = 0)$ $P \mathrel{\mathcal{R}} Q$ and, by definition of ready simulation, $P \stackrel{\omega}{\not\longrightarrow}$ implies $Q \stackrel{\omega}{\not\longrightarrow}$.

$(n = m + 1)$ Let $P = S_0 \xrightarrow{\tau} S_1 \xrightarrow{\tau} S_2 \ldots \xrightarrow{\tau} S_{m+1}$ and $S_i \stackrel{\omega}{\not\longrightarrow}$ for $0 \leq i \leq m + 1$. By the inductive hypothesis, $Q = S'_0 \xrightarrow{\tau} S'_1 \xrightarrow{\tau} S'_2 \ldots \xrightarrow{\tau} S'_m$, $S'_i \stackrel{\omega}{\not\longrightarrow}$ and $S_i \mathrel{\mathcal{R}} S'_i$ for $0 \leq i \leq m$. It follows from $S_m \xrightarrow{\tau} S_{m+1}$ and $S_m \mathrel{\mathcal{R}} S'_m$ that $S'_m \xrightarrow{\tau} S'_{m+1}$, for some S'_{m+1} such that $S_{m+1} \mathrel{\mathcal{R}} S'_{m+1}$. $\mathcal{R}$ is a ready simulation, hence $S_{m+1} \stackrel{\omega}{\not\longrightarrow}$ implies $S'_{m+1} \stackrel{\omega}{\not\longrightarrow}$.

□

Theorem 3.7. *Let P, Q be processes in an image-finite LTS $\mathcal{T} = \langle St, \mathscr{N}, \longrightarrow \rangle$. If $P \precsim^r Q$ then $Q \sqsubseteq^{\mathsf{HOLn}}_{\mathrm{must}} P$.*

Proof. Suppose that $P \precsim^r Q$ and that P **must pass** C does not hold, for some HOLn-context C. By Theorem 3.4, $C[P] \precsim^r C[Q]$, which implies that there is a ready simulation $\mathcal{R}$ defined over the set of actions $\mathscr{A}^*_{\tau,\omega}$ such that $C[P] \mathrel{\mathcal{R}} C[Q]$. There are two cases.

- For some $n \in \mathbb{N}$, $C[P] = S_0 \xrightarrow{\tau} S_1 \xrightarrow{\tau} S_2 \ldots \xrightarrow{\tau} S_n \stackrel{\tau}{\not\longrightarrow}$ and $S_i \stackrel{\omega}{\not\longrightarrow}$ for $0 \leq i \leq n$. By Lemma 3.6, there are $S'_0, \ldots, S'_n$ such that $C[Q] = S'_0 \xrightarrow{\tau} S'_1 \xrightarrow{\tau} S'_2 \ldots \xrightarrow{\tau} S'_n$, $S'_i \stackrel{\omega}{\not\longrightarrow}$ and $S_i \mathrel{\mathcal{R}} S'_i$ for $0 \leq i \leq n$. It follows from $S_n \stackrel{\tau}{\not\longrightarrow}$ and $S_n \mathrel{\mathcal{R}} S'_n$ that $S'_n \stackrel{\tau}{\not\longrightarrow}$. Therefore, Q **must pass** C does not hold.

- There is an infinite path $C[P] = S_0 \xrightarrow{\tau} S_1 \xrightarrow{\tau} S_2 \ldots$ such that $S_i \stackrel{\omega}{\not\longrightarrow}$ for all $i \geq 0$. By Lemma 3.6, for every $n \in \mathbb{N}$ there is a τ-labelled path $C[Q] = S'_0 \xrightarrow{\tau} S'_1 \xrightarrow{\tau} S'_2 \ldots \xrightarrow{\tau} S'_n$ such that $S'_i \stackrel{\omega}{\not\longrightarrow}$ and $S_i \mathrel{\mathcal{R}} S'_i$ for $0 \leq i \leq n$. If $S_n \xrightarrow{\tau} S_{n+1}$ then there is an S'_{n+1} such that $S'_n \xrightarrow{\tau} S'_{n+1}$ and $S_{n+1} \mathrel{\mathcal{R}} S'_{n+1}$, which implies $S'_{n+1} \stackrel{\omega}{\not\longrightarrow}$. The path from $C[P]$ is infinite, so this procedure can be repeated infinitely many times: we start from some $n \in \mathbb{N}$ and we build an infinite τ-labelled path from $C[Q]$ such that no one state in the path performs ω. Hence, Q **must pass** C does not hold in this case as well.

□

3.3.3 Characterization of *may*-equivalence

We define a function translating RSL-formulas to HOLn-processes. Theorem 3.8 shows that this encoding is correct and allows us to prove that *may*-equivalence coincides with ready simulation equivalence on nondeterministic processes (Theorem 3.9).[38]

Let L be an infinite subset of $\mathscr{N}$ and let $l \in \mathscr{N}$ be an atomic action which is not in L. The function $[\![\cdot]\!]^l_L$ mapping an RSL-formula F over $\mathscr{N}$ to a process in HOLn is defined by structural induction on F:

$$\begin{aligned}
&[\![\top]\!]^l_L = \omega\\
&[\![\neg a]\!]^l_L = \widetilde{a}^{\rangle l}.\omega\\
&[\![\langle\, a\,\rangle F]\!] = \bar{a}^{\rangle l}.[\![F]\!]^l_L\\
&[\![F_1 \wedge F_2]\!]^l_L = \mathtt{pass}(x)_l.\langle x\rangle_{l_1} \mid [\![F_1]\!]^{l_1}_{L_1}\{{}^{l_1}\!/\!_{\omega}\} \mid \bar{l}_1.(\langle x\rangle_2 \mid [\![F_2]\!]^{l_2}_{L_2}\{{}^{l_2}\!/\!_{\omega}\}) \mid \bar{l}_2.\omega
\end{aligned}$$

where $\{l_1, l_2\} \subseteq L$ and L_1, L_2 are pairwise disjoint infinite subsets of $L\backslash\{l_1, l_2\}$.
Let $C^{l,L}_F$ denote the HOLn-context $\langle[\cdot]\rangle_l \mid [\![F]\!]^l_L$.

Theorem 3.8. *Let $\mathcal{T} = \langle St, \mathscr{N}, \longrightarrow\rangle$ be an image-finite LTS. For every process P in $\mathcal{T}$ and for every RSL-formula F, $P \models F$ if and only if $C^{l,L}_F[P] \Longrightarrow S \xrightarrow{\omega}$, for some HOLn-process S.*

Proof. By induction on F.

$(F = \top)$ $\top$ is always true at P and $C^{l,L}_F[P] = \langle P\rangle_l \mid \omega \xrightarrow{\omega}$.

$(F = \neg a)$ If $P \models \neg a$ then $P \stackrel{a}{\not\longrightarrow}$ and we have that:

$$\frac{\dfrac{P \stackrel{a}{\not\longrightarrow}}{\langle P\rangle_l \xrightarrow{\bar{\widetilde{a}}^{\rangle l}} \mathbf{0}}\,(\text{Oref}) \qquad \dfrac{}{\widetilde{a}^{\rangle l}.\omega \xrightarrow{\widetilde{a}^{\rangle l}} \omega}\,(\text{Iref})}{C^{l,L}_{\neg a}[P] \xrightarrow{\tau} \langle \mathbf{0}\rangle_l \mid \omega}\,(\text{sync})$$

If $P \not\models \neg a$ then $P \xrightarrow{a}$ and $\widetilde{a}^{\rangle l}.\omega$ cannot synchronize with $\langle P\rangle_l$. Moreover, P does not perform τ-labelled transitions, so $C^{l,L}_{\neg a}[P]$ never reaches a successful state through a τ-labelled path.

$(F = \langle\, a\,\rangle F')$ $P \models \langle\, a\,\rangle F'$ implies that $P' \models F'$, for some P' such that $P \xrightarrow{a} P'$. Hence, we have the following derivation:

[38] This result is inspired by the work of Bard Bloom, Sorin Istrail and Albert R. Meyer (Bloom, Istrail, and Meyer 1995). The authors defined the family of GSOS rule systems, where processes can be copied and actions can be refused, and proved that two processes P and Q are ready simulation equivalent if and only if $C[P]$ and $C[Q]$ are trace equivalent for every GSOS-context C.

$$\dfrac{\dfrac{P \xrightarrow{a} P'}{\langle P\rangle_l \xrightarrow{a^{\rangle l}} \langle P'\rangle_l}\,(\text{kell}) \qquad \dfrac{}{\bar{a}^{\rangle l}.[\![F']\!]^l_L \xrightarrow{\bar{a}^{\rangle l}} [\![F']\!]^l_L}\,(\text{pref})}{C^{l,L}_F[P] \xrightarrow{\tau} \langle P'\rangle_l \,|\, [\![F']\!]^l_L}\,(\text{sync})$$

By the inductive hypothesis, $P' \models F'$ implies $\langle P'\rangle_l \,|\, [\![F']\!]^l_L \Longrightarrow S \xrightarrow{\omega}$ for some HOLn-process S. Therefore, $C^{l,L}_F[P] \Longrightarrow S \xrightarrow{\omega}$.

Suppose now that $P \not\models \langle\, a \,\rangle F'$. If $P \not\xrightarrow{a}$ then, symmetrically to the previous case (the case when $P \not\models \neg a$), we have that $C^{l,L}_F[P] \not\xrightarrow{\tau}$ and $C^{l,L}_F[P] \not\xrightarrow{\omega}$. If $P \xrightarrow{a} P'$, then $P' \not\models F'$. As above, we derive that $C^{l,L}_F[P] \xrightarrow{\tau} \langle P'\rangle_l \,|\, [\![F']\!]^l_L$, but it follows from the inductive hypothesis on F' that $C^{l,L}_{F'}[P'] = \langle P'\rangle_l \,|\, [\![F']\!]^l_L$ never reaches a state performing an ω-labelled transition.

$(F = F_1 \wedge F_2)$ We have that:

$$C^{l,L}_F[P'] = \langle P\rangle_l \,|\, [\![F_1 \wedge F_2]\!]^l_L = \langle P\rangle_l \,|\, \texttt{pass}(x)_l.(C^{l_1,L_1}_{F_1}[x]\{^{l_1}/_{\omega}\} \,|\, \bar{l}_1.(C^{l_2,L_2}_{F_2}[x]\{^{l_2}/_{\omega}\}) \,|\, \bar{l}_2.\omega)$$

and there is one and only one τ-labelled transition that $C^{l,L}_F[P']$ can perform, i.e. the transition derived as follows:

$$\dfrac{\dfrac{}{\langle P\rangle_l \xrightarrow{\texttt{pass}(P)_l} \mathbf{0}}\,(\text{Opass}) \qquad \dfrac{}{[\![F_1 \wedge F_2]\!]^l_L \xrightarrow{\texttt{pass}(x)_l} C^{l_1,L_1}_{F_1}[x]\{^{l_1}/_{\omega}\} \,|\, \bar{l}_1.(C^{l_2,L_2}_{F_2}[P]\{^{l_2}/_{\omega}\}) \,|\, \bar{l}_2.\omega}\,(\text{Ipass})}{C^{l,L}_F[P'] \xrightarrow{\tau} \mathbf{0} \,|\, C^{l_1,L_1}_{F_1}[P]\{^{l_1}/_{\omega}\} \,|\, \bar{l}_1.(C^{l_2,L_2}_{F_2}[P]\{^{l_2}/_{\omega}\}) \,|\, \bar{l}_2.\omega}\,(\text{psync})$$

Let $A = \mathbf{0} \,|\, C^{l_1,L_1}_{F_1}[P]\{^{l_1}/_{\omega}\} \,|\, \bar{l}_1.(C^{l_2,L_2}_{F_2}[P]\{^{l_2}/_{\omega}\}) \,|\, \bar{l}_2.\omega$. If $P \models F$ then $P \models F_1$ and $P \models F_2$, which implies by the inductive hypothesis that there are HOLn-processes S_1, S'_1, S_2, S'_2 such that $C^{l_1,L_1}_{F_1}[P] \Longrightarrow S_1 \xrightarrow{\omega} S'_1$ and $C^{l_2,L_2}_{F_2}[P] \Longrightarrow S_2 \xrightarrow{\omega} S'_2$. Therefore,

$$\begin{aligned}
C^{l,L}_F[P'] \xrightarrow{\tau} A &\Longrightarrow \mathbf{0} \,|\, S_1\{^{l_1}/_{\omega}\} \,|\, \bar{l}_1.(C^{l_2,L_2}_{F_2}[P]\{^{l_2}/_{\omega}\}) \,|\, \bar{l}_2.\omega \\
&\xrightarrow{\tau} \mathbf{0} \,|\, S'_1\{^{l_1}/_{\omega}\} \,|\, C^{l_2,L_2}_{F_2}[P]\{^{l_2}/_{\omega}\} \,|\, \bar{l}_2.\omega \\
&\Longrightarrow \mathbf{0} \,|\, S'_1\{^{l_1}/_{\omega}\} \,|\, S_2\{^{l_2}/_{\omega}\} \,|\, \bar{l}_2.\omega \\
&\xrightarrow{\tau} \mathbf{0} \,|\, S'_1\{^{l_1}/_{\omega}\} \,|\, S'_2\{^{l_2}/_{\omega}\} \,|\, \omega \\
&\xrightarrow{\omega}
\end{aligned}$$

Finally, suppose that $P \not\models F_1 \wedge F_2$. By the inductive hypothesis, at least one of the processes $C^{l_1,L_1}_{F_1}[P], C^{l_2,L_2}_{F_2}[P]$ cannot reach a state performing ω by means of a τ-labelled path. If $C^{l_1,L_1}_{F_1}[P]$ fails to reach a successful state, then $C^{l,L}_F[P] \Longrightarrow S$ implies that $S = \mathbf{0} \,|\, S_1 \,|\, \bar{l}_1.C^{l_2,L_2}_{F_2}[P]\{^{l_2}/_{\omega}\} \,|\, \bar{l}_2.\omega$, where $S_1 \not\xrightarrow{l_1}$. Otherwise, if $C^{l_1,L_1}_{F_1}[P] \Longrightarrow S_1 \xrightarrow{\omega}$

then $C_{F_2}^{l_2,L_2}[P]$ fails. In this case, $C_F^{l,L}[P] \Longrightarrow S$ implies that $S \overset{l_2}{\not\longrightarrow}$. So, it holds that $S \overset{\omega}{\not\longrightarrow}$ for every S such that $C_F^{l,L}[P] \Longrightarrow S$.

□

Theorem 3.9. *Let P, Q be processes in an image-finite LTS $\mathcal{T} = \langle St, \mathcal{N}, \longrightarrow\rangle$. $P \eqsim^r Q$ if and only if $P \simeq_{\text{may}}^{\mathsf{HOLn}} Q$.*

Proof.

($\Rightarrow$) The result follows from Theorem 3.5 and from the fact that $\eqsim^r$ is a symmetric relation.

($\Leftarrow$) Suppose that $P \not\lesssim^r Q$. It follows from Theorem 1.38 that there is a formula F of Ready Simulation Logic such that $P \models F$ and $Q \not\models F$. By Theorem 3.8, $C_F^{l,L}[P] \Longrightarrow S \xrightarrow{\omega}$ for some HOLn-process S, while $Q \Longrightarrow S'$ implies $S' \overset{\omega}{\not\longrightarrow}$. Therefore, there is a HOL-context $C = C_F^{l,L}$ such that $\top \in \mathbb{S}(C_F^{l,L}[P])$ and $\top \notin \mathbb{S}(C_F^{l,L}[Q])$, which implies that $P \not\sqsubseteq_{\text{may}}^{\mathsf{HOLn}} Q$. Symmetrically, if $Q \not\lesssim^r P$ then $Q \not\sqsubseteq_{\text{may}}^{\mathsf{HOLn}} P$ and the result follows by contraposition.

□

3.3.4 Characterization of testing equivalences

The results proved in the previous sections are summarized by Theorem 3.10: in the testing scenario where the tests are HOL-contexts and the tested processes are nondeterministic processes it holds that ready simulation equivalence coincides with both *may*-equivalence and *test*-equivalence. As a corollary, we have that *must*-equivalence is included in all these equivalences.

Theorem 3.10. *Let P, Q be processes in an image-finite LTS $\mathcal{T} = \langle St, \mathcal{N}, \longrightarrow\rangle$. The following statements are equivalent.*

1. $P \eqsim^r Q$
2. $P \simeq_{\text{may}}^{\mathsf{HOLn}} Q$
3. $P \simeq_{\text{test}}^{\mathsf{HOLn}} Q$.

Proof. (1) and (2) are equivalent by Theorem 3.9. By Theorem 3.7, (1) implies (3). By the definition of $\simeq_{\text{test}}^{\mathsf{HOLn}}$, (3) implies (2).

□

3.4 Testing probabilistic reactive processes

Analogously to what we did in Section 3.3, we extend HOL to a new language by adding to it an infine set of constant's names. This time, we interpret these constants as reactive probabilistic processes. Since we want these processes to satisfy the minimal deviation assumption with respect to some minimal probability value π, we fix this minimal value from the beginning and we let $Pr(\mathsf{LS})$ denote the class of probabilistic reactive processes satisfying the minimal deviation assumption with respect to π.

Definition 3.11. The terms of the language HOLp on $\mathscr{A}$ are defined by the following grammar:

$$P ::= \mathbf{0} \;\Big|\; x \;\Big|\; S \;\Big|\; \alpha^*.P \;\Big|\; \widetilde{\alpha}^{\rangle l}.P \;\Big|\; \omega.P \;\Big|\; P \,|\, P \;\Big|\; \langle P \rangle_l \;\Big|\; \mathtt{pass}(x)_l.P$$

where S is (the name of) a process in the class $Pr(\mathsf{LS})$ of reactive probabilistic processes satisfying the minimal deviation assumption with respect to the minimal probability value π.
The set $Pr(\mathsf{HOLp})$ of HOLp-processes is the set of HOLp's closed terms.

In what follows, we say that P is an LS-process (a "Larsen and Skou's process") if P is a reactive probabilistic process in $Pr(\mathsf{LS})$.
Let C be a HOLp-context where n distinct holes appear and let $P_1, ..., P_n$ be HOLp-terms. We say that C is an n-ary HOLp-context and we write $C[P_1, ..., P_n]$ to denote the HOLp-term obtained by substituting P_i to the i-th occurrence of $[\cdot]$ in C, for $1 \leq i \leq n$.
For any $l \in \mathscr{N}$, $P' \in \mathsf{HOLp}$ and $\Delta_1, \Delta_2 \in \mathcal{D}(\mathsf{HOLp})$ we define the distributions $\Delta_1 \,|\, \Delta_2$, $\langle \Delta_1 \rangle_l$ and $\Delta_1 \,|\, \Delta_2\{{}^{P'}\!/\!{}_x\}$ on HOLp-terms as follows:

$$\Delta_1 \,|\, \Delta_2(P) = \begin{cases} \Delta_1(P_1) \cdot \Delta_2(P_2) & \text{if } P = P_1 \,|\, P_2 \\ 0 & \text{else} \end{cases}$$

$$\langle \Delta_1 \rangle_l(P) = \begin{cases} \Delta_1(P') & \text{if } P = \langle P' \rangle_l \\ 0 & \text{else} \end{cases}$$

$$\Delta_1 \,|\, \Delta_2\{{}^{P'}\!/\!{}_x\}(P) = \Delta_1 \,|\, \Delta_2(P\{{}^{P'}\!/\!{}_x\}).$$

The operational semantic of the language HOLp on $\mathscr{N}$ is defined by the LTS

$$\langle Pr(\mathsf{HOLp}), \mathscr{A}^{\mathsf{HOL}}, \longrightarrow \rangle,$$

where $\mathscr{A}^{\mathsf{HOL}} = \mathscr{A}^*_{\tau,\omega} \cup \widetilde{\mathscr{N}^*} \cup \{\mathtt{pass}(\rho)_l |\; \rho \in Pr(\mathsf{HOLp}) \cup \mathcal{V} \text{ and } l \in \mathscr{N}\}$ and the transition relation is given by the rules in figure 3.2.

$$\frac{S \xrightarrow{a}{}' \Delta}{S \xrightarrow{a} \Delta}\ (LS)$$

where $\longrightarrow'$ is the transition relation of the pLTS for S

$$\frac{}{\mu.P \xrightarrow{\mu} \overline{P}}\ (\text{pref})_{\mu \in \mathscr{A}^\star \cup \{\omega\}}$$

$$\frac{}{\widetilde{\alpha}\rangle^l.P \xrightarrow{\widetilde{\alpha}\rangle^l} \overline{P}}\ (\text{Iref})$$

$$\frac{P \overset{\alpha}{\not\longrightarrow} \text{ and } P \text{ is stable}}{\langle P \rangle_l \xrightarrow{\overline{\widetilde{\alpha}}\rangle^l} \overline{\mathbf{0}}}\ (\text{Oref})$$

$$\frac{P_1 \xrightarrow{\mu} \Delta_1}{P_1 \,|\, P_2 \xrightarrow{\mu} \Delta_1 \,|\, \overline{P_2}}\ (\text{parL})_{\mu \in \mathscr{A}^{\mathsf{HOLp}}}$$

$$\frac{P_1 \xrightarrow{\mu} \Delta_1}{P_2 \,|\, P_1 \xrightarrow{\mu} \overline{P_2} \,|\, \Delta_1}\ (\text{parR})_{\mu \in \mathscr{A}^{\mathsf{HOLp}}}$$

$$\frac{P_1 \xrightarrow{\mu} \Delta_1 \qquad P_2 \xrightarrow{\bar{\mu}} \Delta_2}{P_1 \,|\, P_2 \xrightarrow{\tau} \Delta_1 \,|\, \Delta_2}\ (\text{sync})_{\mu \in \mathscr{A}^\star \cup \widetilde{\mathscr{A}}^\star}$$

$$\frac{P \xrightarrow{\alpha} \Delta}{\langle P \rangle_l \xrightarrow{\alpha\rangle^l} \langle \Delta \rangle_l}\ (\text{kell})$$

$$\frac{P \xrightarrow{\tau} \Delta}{\langle P \rangle_l \xrightarrow{\tau} \langle \Delta \rangle_l}\ (\tau\text{kell})$$

$$\frac{}{\mathsf{pass}(x)_l.P \xrightarrow{\mathsf{pass}(x)_l} \overline{P}}\ (\text{Ipass})$$

$$\frac{}{\langle P \rangle_l \xrightarrow{\mathsf{pass}(P)_l} \overline{\mathbf{0}}}\ (\text{Opass})$$

$$\frac{P_1 \xrightarrow{\mathsf{pass}(\rho_1)_l} \Delta_1 \qquad P_2 \xrightarrow{\mathsf{pass}(\rho_2)_l} \Delta_2}{P_1 \,|\, P_2 \xrightarrow{\tau} \Delta_1 \,|\, \Delta_2\{{}^{P}\!/x\}}\ (\text{psync})$$

where either $\rho_1 = P$ and $\rho_2 = x$ or $\rho_1 = x$ and $\rho_2 = P$

Figure 3.2: SOS for processes in HOLp.

3.4.1 Testing preorders on probabilistic processes

We now consider the testing scenario $\langle Pr(\mathsf{LS}), Ctx(\mathsf{HOL}), [0,1], Succ\rangle$, where the application of a test C to an LS-process P is the substitution of P to the holes in C, i.e. there is a function $\mathbb{S} : Pr(\mathsf{HOLp}) \to \mathscr{P}^+([0,1])$ such that $Succ(C,P) = \mathbb{S}(C[P])$. The problem in defining $\mathbb{S}$ is that, in general, a HOLp process $C[P]$, where $C \in Ctx(\mathsf{HOL})$ and $P \in Pr(\mathsf{LS})$, is not a reactive probabilistic process. Therefore, we introduce oracles, i.e. schedulers which allow the resolution of the nondeterministic choices in the probabilistic

Labelled Transition System for HOLp.

Definition 3.12. Let $\mathscr{O}$ be a function assigning a probability distribution on $Pr(\mathsf{HOLp})$ to every n-tuple of HOLp-processes $< P_1, P_2, ..., P_n >$, for all $n \in \mathbb{N}$. We say that $\mathscr{O}$ is an *oracle* on $Pr(\mathsf{HOLp})$ if $\mathscr{O}(< P_1, ..., P_n >) = \Delta$ implies that $P_n \xrightarrow{\tau} \Delta$.

For a given HOLp-process P, we define by induction on $n \in \mathbb{N}$ when $< P_1, ..., P_n > \in path(\mathscr{O}, P)$:

- $< P > \in path(\mathscr{O}, P)$,
- if $< P_1, ..., P_n > \in path(\mathscr{O}, P)$, $P_n \xrightarrow{\tau}$ and $P' \in \lceil \mathscr{O}(< P_1, ..., P_n >) \rceil$ then $< P_1, ..., P_n, P' > \in path(\mathscr{O}, P)$.

The set $\omega path(\mathscr{O}, P)$ of all the successful computations from P with respect to $\mathscr{O}$ is the set:

$$\{< P_1, ..., P_n > \,|\, n \in \mathbb{N}, < P_1, ..., P_n > \in path(\mathscr{O}, P), P_n \xrightarrow{\omega} \text{ and } P_i \overset{\omega}{\not\longrightarrow} \text{ for } 1 \leq i < n\}.$$

The function $\mathbb{S}^{\mathscr{O}} : Pr(\mathsf{HOLp}) \to [0, 1]$ associates to a process P its probability of reaching a successful state, relatively to the oracle $\mathscr{O}$:

$$\mathbb{S}^{\mathscr{O}}(P) = \sum_{<P_1,...,P_n> \in \, \omega path(\mathscr{O},P)} \prod_{i=1}^{n-1} \mathscr{O}(P_i)(P_{i+1})$$

where $\mathbb{S}^{\mathscr{O}}(P) = 0$ if $\omega path(\mathscr{O}, P) = \emptyset$.

The *set of probabilities of success* of P is obtained by considering all possible oracles on $Pr(\mathsf{HOLp})$:

$$\mathbb{S}(P) = \{\mathbb{S}^{\mathscr{O}}(P) | \, \mathscr{O} \text{ is an oracle on } Pr(\mathsf{HOLp})\}.$$

We can now restate the testing preorders defined in Section 3.1 as follows:

$$\begin{array}{lll} P \sqsubseteq^{\mathsf{HOLp}}_{\text{may}} Q \text{ iff} & \forall C \in Ctx(\mathsf{HOL}) & \bigsqcup \mathbb{S}(C[P]) \leq \bigsqcup \mathbb{S}(C[Q]) \\ P \sqsubseteq^{\mathsf{HOLp}}_{\text{must}} Q \text{ iff} & \forall C \in Ctx(\mathsf{HOL}) & \bigsqcap \mathbb{S}(C[P]) \leq \bigsqcap \mathbb{S}(C[Q]) \\ P \sqsubseteq^{\mathsf{HOLp}}_{\text{test}} Q \text{ iff} & P \sqsubseteq^{\mathbf{T}}_{\text{may}} Q \text{ and } P \sqsubseteq^{\mathbf{T}}_{\text{must}} Q. & \end{array}$$

Given a class $Pr(\mathsf{LS})$ of LS-processes, we prove (Theorem 3.29) that probabilistic bisimilarity on this class of processes coincides with both *may*-equivalence and *must*-equivalence in the testing scenario $\langle Pr(\mathsf{LS}), Ctx(\mathsf{HOL}), [0, 1], Succ \rangle$ defined above.

The following two sections are devoted to the proof of this result. In Section 3.4.2 we show that if P and Q are LS-processes which are not probabilistically bisimilar, then there is a HOL-context C such that $C[P]$ and $C[Q]$ are neither *may*-equivalent nor *must*-equivalent. In Section 3.4.3 we prove the opposite implication: probabilistically bisimilar LS-processes are *test*-equivalent.

3.4.2 HOL-contexts discriminate non-probabilistically bisimilar processes

Let L be an infinite subset of $\mathscr{N}$ and $l \notin L$. The function $[\![\cdot]\!]^l_L$ mapping pairs $< t, e >$, where t is a test in $\mathbf{T}^{\mathsf{LS}}$ defined over the set of actions $\mathscr{N}$ and $e \in O_t$, to processes in HOLp is defined by structural induction on t:

$$
\begin{aligned}
&[\![< \omega, 1_\omega >]\!]^l_L = \omega\\
&[\![< a.t, 0_a >]\!]^l_L = \widetilde{a}^{\rangle l}.\omega\\
&[\![< a.t, 1_a : e >]\!] = \bar{a}^{\rangle l}.[\![< t, e >]\!]^l_L\\
&[\![< (t_1, ..., t_n), (e_1, ..., e_n) >]\!]^l_L =\\
&\quad = \mathtt{pass}(x)_l.\langle x\rangle_{l_1} \,|\, [\![< t_1, e_1 >]\!]^{l_1}_{L_1}\{{}^{l_1}\!/\omega\} \,|\, \mathop{\Big|\Big|}_{i=2}^{n} (\bar{l}_{i-1}.(\langle x\rangle_{l_i} \,|\, [\![< t_i, e_i >]\!]^{l_i}_{L_i}\{{}^{l_i}\!/\omega\})) \,|\, \bar{l}_n.\omega
\end{aligned}
$$

where $\{l_1, l_2\} \subseteq L$ and $L_1, ..., L_n$ are pairwise disjoint infinite subsets of $L\backslash\{l_1, l_2\}$.
Let $C^{l,L}_{<t,e>}$ denote the HOL-context $\langle[\cdot]\rangle_l \,|\, [\![< t, e >]\!]^l_L$. We want to show that the process $C^{l,L}_{<t,e>}[P]$ (i.e. the process obtained by filling the hole of $C^{l,L}_{<t,e>}$ with an LS-process) is suitable for playing the role that $< t, e >$ plays when applied to P in the Larsen and Skou's testing scenario.
As illustrated by Example 3.13, it turns out that $C^{l,L}_{<t,e>}[P]$ could be a nondeterministic probabilistic process.

Example 3.13. Consider the following translation:

$$
\begin{aligned}
&[\![< (a.\omega, b.\omega), (1_a : 1_\omega, 1_b : 1_\omega) >]\!]^l_L =\\
&= \mathtt{pass}(x)_l.\langle x\rangle_{l_1} \,|\, [\![< a.\omega, 1_a : 1_\omega >]\!]^{l_1}_{L_1}\{{}^{l_1}\!/\omega\} \,|\, \bar{l}_1.(\langle x\rangle_{l_2} \,|\, [\![< b.\omega, 1_b : 1_\omega >]\!]^{l_2}_{L_2}\{{}^{l_2}\!/\omega\}) \,|\, \bar{l}_2.\omega =\\
&= \mathtt{pass}(x)_l.\langle x\rangle_{l_1} \,|\, \bar{a}^{\rangle l_1}.\omega \,|\, \bar{l}_1.(\langle x\rangle_{l_2} \,|\, \bar{b}^{\rangle l_2}.\omega) \,|\, \bar{l}_2.\omega
\end{aligned}
$$

and the LS-process P such that $P \xrightarrow{a} \overline{P}$ and $P \xrightarrow{b} \overline{P}$. We have

$$C^{l,L}_{<(a.\omega,b.\omega),(1_a:1_\omega,1_b:1_\omega)>}[P] \xrightarrow{\tau} \overline{\langle P\rangle_{l_1} \,|\, \bar{a}^{\rangle l_1}.l_1 \,|\, \bar{l}_1.(\langle P\rangle_{l_2} \,|\, \bar{b}^{\rangle l_2}.\omega) \,|\, \bar{l}_2.\omega \,|\, \mathbf{0}}$$

and

$$\langle P\rangle_{l_1} \,|\, \bar{a}^{\rangle l_1}.l_1 \,|\, \bar{l}_1.(\langle P\rangle_{l_2} \,|\, \bar{b}^{\rangle l_2}.\omega) \,|\, \bar{l}_2.\omega \,|\, \mathbf{0} \xrightarrow{\bar{l}_1} \overline{\langle P\rangle_{l_1} \,|\, \bar{a}^{\rangle l_1}.l_1 \,|\, \langle P\rangle_{l_2} \,|\, \bar{b}^{\rangle l_2}.\omega \,|\, \bar{l}_2.\omega \,|\, \mathbf{0}},$$

where $\langle P\rangle_{l_1} \,|\, \bar{a}^{\rangle l_1}.l_1 \,|\, \langle P\rangle_{l_2} \,|\, \bar{b}^{\rangle l_2}.\omega \,|\, \bar{l}_2.\omega \,|\, \mathbf{0}$ can perform a τ-labelled transition both to the distribution $\overline{\langle P\rangle_{l_1} \,|\, l_1 \,|\, \langle P\rangle_{l_2} \,|\, \bar{b}^{\rangle l_2}.\omega \,|\, \bar{l}_2.\omega \,|\, \mathbf{0}}$ and to $\overline{\langle P\rangle_{l_1} \,|\, \bar{a}^{\rangle l_1}.l_1 \,|\, \langle P\rangle_{l_2} \,|\, \omega \,|\, \bar{l}_2.\omega \,|\, \mathbf{0}}$.

Nevertheless, we can prove that the process $C^{l,L}_{<t,e>}[P]$ is τ-reactive, that is, $C^{l,L}_{<t,e>}[P]$ is reactive if we consider only the τ-labelled paths from $C^{l,L}_{<t,e>}[P]$.

Definition 3.14. The relation $\Longrightarrow$ on HOLp-processes is the smallest relation closed under the following rules:

$$\frac{}{P \Longrightarrow P} \qquad\qquad \frac{P \xrightarrow{\tau} \Delta \qquad P' \in \lceil \Delta \rceil \qquad P' \Longrightarrow P''}{P \Longrightarrow P''}$$

We say that a process P is *τ-reactive* if whenever $P \Longrightarrow P'$ we have that:

$$P' \xrightarrow{\tau} \Delta_1 \text{ and } P' \xrightarrow{\tau} \Delta_2 \text{ implies } \Delta_1 = \Delta_2$$

for all $P' \in \mathsf{HOLp}$ and $\Delta_1, \Delta_2 \in \mathcal{D}(\mathsf{HOLp})$.

Theorem 3.15. *Let P be an LS-process. The HOLp-process $C^{l,L}_{<t,e>}[P]$ is a τ-reactive probabilistic process.*

Proof. We prove by induction on $< t, e >$ that for every $S \in Pr(\mathsf{HOLp})$ such that $C^{l,L}_{<t,e>}[P] \Longrightarrow S$ the following properties hold:

(1) if $S \xrightarrow{\nu}$ then either $\nu \in L \,\cup\, \overline{L} \,\cup\, \{\omega, \tau\}$ or ν is of the form $\alpha^{\rangle h}, \widetilde{\alpha}^{\rangle h}, \overline{\widetilde{\alpha}}^{\rangle h}, \mathtt{pass}(x)_h$ or $\mathtt{pass}(Q)_h$, where $h \in L \cup \{l\}$,

(2) if $S \xrightarrow{\omega} \Delta_1$ and $S \xrightarrow{\omega} \Delta_2$ then $\Delta_1 = \Delta_2$, $S \stackrel{\tau}{\not\longrightarrow}$ and for every $S' \in \lceil \Delta_1 \rceil$, $S' \stackrel{\tau}{\not\longrightarrow}$ and $S' \stackrel{\omega}{\not\longrightarrow}$,

(3) if $S \xrightarrow{\tau} \Delta_1$ and $S \xrightarrow{\tau} \Delta_2$ then $\Delta_1 = \Delta_2$.

The proof is by structural induction on $< t, e >$. We only consider the two non-trivial cases:

($< t, e > = < a.t', 1_a : e' >$ *and* $P \xrightarrow{a} \Delta$) If $S = C^{l,L}_{<t,e>}[P]$, then the result follows from the fact that P is an LS-process, so $P \stackrel{\tau}{\not\longrightarrow}$, $P \stackrel{\omega}{\not\longrightarrow}$ and there is one and only one τ-labelled transition from $C^{l,L}_{<t,e>}[P]$, i.e. the transition derived as follows:

$$\dfrac{\dfrac{P \xrightarrow{a} \Delta}{\langle P \rangle_l \xrightarrow{a^{\rangle l}} \langle \Delta \rangle_l}\,(\text{kell}) \qquad \dfrac{}{\bar{a}^{\rangle l}.[\![< t', e' >]\!]^l_L \xrightarrow{\bar{a}^{\rangle l}} \overline{[\![< t', e' >]\!]^l_L}}\,(\text{pref})}{C^{l,L}_{<t,e>}[P] \xrightarrow{\tau} \langle \Delta \rangle_l \,|\, \overline{[\![< t', e' >]\!]^l_L}}\,(\text{sync})$$

Suppose that $C^{l,L}_{<t,e>}[P] \xrightarrow{\tau} \Delta'$ and for some $S' \in \Delta'$, $S' \Longrightarrow S$. Then $\Delta' = \langle \Delta \rangle_l \,|\, \overline{[\![< t', e' >]\!]^l_L}$ and $S' = C^{l,L}_{<t',e'>}[P']$, for some $P' \in \lceil \Delta \rceil$. The result follows from the inductive hypothesis.

$(< t, e > = < (t_1, ..., t_n), (e_1, ..., e_n) >)$ By induction on n.

If $n = 2$ there is one and only one τ-labelled transition from $C^{l,L}_{<t,e>}[P]$, leading to the probability distribution:

$$\overline{\langle P\rangle_{l_1} \mid [\![< t_1, e_1 >]\!]^{l_1}_{L_1}\{^{l_1}\!/\!_{\omega}\} \mid \bar{l}_1.(\langle P\rangle_{l_2} \mid [\![< t_2, e_2 >]\!]^{l_2}_{L_2}\{^{l_2}\!/\!_{\omega}\}) \mid \bar{l}_2.\omega \mid \mathbf{0}}$$

which is equal to $\overline{C^{l_1,L_1}_{<t_1,e_1>}[P] \mid \bar{l}_1.C^{l_2,L_2}_{<t_2,e_2>}[P] \mid \bar{l}_2.\omega \mid \mathbf{0}}$. Hence, if $S = C^{l,L}_{<t,e>}[P]$ then the conditions (1) and (2) immediately follow; condition (3) holds because of the inductive hypothesis on $C^{l_1,L_1}_{<t_1,e_1>}[P]$ and because of the fact that $L_1 \cup \{l_1, \bar{l}_1, l_2, \bar{l}_2\} \subseteq L$.

Let $A = C^{l_1,L_1}_{<t_1,e_1>}[P] \mid \bar{l}_1.C^{l_2,L_2}_{<t_2,e_2>}[P] \mid \bar{l}_2.\omega \mid \mathbf{0}$. If $C^{l,L}_{<t,e>}[P] \longrightarrow \Delta'$ and $S' \Longrightarrow S$, for some $S' \in \Delta'$, then $\Delta' = \overline{A}$, where $A \overset{\omega}{\not\longrightarrow}$.

$A \Longrightarrow S$ implies that S satisfies one of the following conditions:

(a) the subprocess $C^{l_1,L_1}_{<t_1,e_1>}[P]$ of S has not yet reported success, so $S = S_1 \mid S_2$, where $C^{l_1,L_1}_{<t_1,e_1>}[P] \Longrightarrow S_1'$, $S_1 = S_1'\{^{l_1}\!/\!_{\omega}\}$ and $S_2 = \bar{l}_1.C^{l_2,L_2}_{<t_2,e_2>}[P] \mid \bar{l}_2.\omega \mid \mathbf{0}$.
The first condition follows immediately from the relative hypothesis of induction, while the second property is vacuously satisfied because $S \overset{\omega}{\not\longrightarrow}$.
If $S_1' \xrightarrow{\tau}$ then $S_1' \overset{\omega}{\not\longrightarrow}$, by the inductive hypothesis of (2). By the inductive hypothesis of (1), the transitions leaving S_1 and S_2 have labels that do not allow synchronizations between the two processes. Therefore, $S \xrightarrow{\tau}$ if and only if $S_1 \xrightarrow{\tau}$ and it follows from the inductive hypothesis of (3) that the third condition holds.
If $S_1' \xrightarrow{\omega} \Delta$ then $S_1' \overset{\tau}{\not\longrightarrow}$, $S_1 \xrightarrow{l_1} \Delta$ and there is one and only one τ-transition from S (as above, this is a consequence of the three hypothesis of induction), i.e. the transition to $\overline{\Delta \mid \bar{l}_1.C^{l_2,L_2}_{<t_2,e_2>}[P] \mid \bar{l}_2.\omega \mid \mathbf{0}}$. Hence, (3) holds in this case too.

(b) the subprocess $C^{l_1,L_1}_{<t_1,e_1>}[P]$ of S reported succes, which implies that $S = S_1 \mid S_2$, where $C^{l_1,L_1}_{<t_1,e_1>}[P] \Longrightarrow S_1'$, $S_1' \xrightarrow{\omega} \Delta$, $S_1 \in \lceil\Delta\rceil$ and $C^{l_2,L_2}_{<t_2,e_2>}[P]\{^{l_2}\!/\!_{\omega}\} \Longrightarrow S_2$.
The proof is similar to the previous one.

The result for the case when $< t, e > = < (t_1, ..., t_{n+1}), (e_1, ..., e_{n+1}) >$ follows analogously by applying the inductive hypothesis on n instead of the one on $< t_2, e_2 >$.

□

If P is a τ-reactive probabilistic process then the function $\mathbb{S}^{\mathcal{O}}$ returns the same probability of success on P for every oracle $\mathcal{O}$ on $Pr(\mathsf{HOLp})$. Thus, oracles are superfluous in this case and $\mathbb{S}(P) = \{\mathbb{S}^{\mathcal{O}}(P)\} = \{\mathbf{S}(P)\}$ for all oracles $\mathcal{O}$ on $Pr(\mathsf{HOLp})$, where:

$$\mathbf{S}(P) = \sum_{<P_1,...,P_n>\in\, \omega path(P)} \prod_{i=1}^{n-1} \Delta_i(P_{i+1})$$

and the set of the successful paths from P is redefined as follows:

$$\begin{aligned}\omega path(P) \quad = \quad & \{< P_1, ..., P_n > \,|\, P_1 = P,\ P_n \xrightarrow{\omega} \text{ and there is a } \Delta_i \text{ such that } P_i \xrightarrow{\tau} \Delta_i, \\ & P_{i+1} \in \lceil \Delta_i \rceil \text{ and } P_i \overset{\omega}{\not\longrightarrow} \text{ for every } i \text{ such that } 1 \leq i \leq n-1\}.\end{aligned}$$

Theorem 3.16. *Let t be a test in $\mathbf{T}^{\mathsf{LS}}$ and $e \in O_t$. For every LS-process P,*

$$\mathcal{P}_{t,P}(e) = \mathbf{S}(C^{l,L}_{<t,e>}[P]).$$

Proof. By structural induction on $< t, e >$.

($< t, e > = < \omega, 1_\omega >$) $\omega path(C^{l,L}_{<t,e>}[P]) = \{< \langle P \rangle_l \,|\, \omega >\}$, so we have that $\mathcal{P}_{\omega,P}(1_\omega) = 1 = \mathbf{S}(C^{l,L}_{<t,e>}[P])$.

($< t, e > = < a.t', 0_a >$) $P \overset{\tau}{\not\longrightarrow}$, hence $C^{l,L}_{<t,e>}[P] \xrightarrow{\tau}$ if and only if $P \overset{a}{\not\longrightarrow}$. If P can perform an a-labelled transition then $\omega path(C^{l,L}_{<t,e>}[P]) = \emptyset$ and $\mathcal{P}_{a.t',P}(0_a) = 0 = \mathbf{S}(C^{l,L}_{<t,e>}[P])$. If $P \overset{a}{\not\longrightarrow}$ then $\langle P \rangle_l \xrightarrow{\overline{\overline{a}\rangle_l}} \mathbf{0}$ and $C^{l,L}_{<t,e>}[P] \xrightarrow{\tau} \overline{\mathbf{0} \,|\, \omega}$. As a consequence, $\omega path(C^{l,L}_{<t,e>}[P]) = \{< C^{l,L}_{<t,e>}[P],\ \mathbf{0} \,|\, \omega >\}$ and both expressions are equal to 1.

($< t, e > = < a.t', 1_a : e' >$) If P cannot perform an a-labelled transition then both expressions are equal to 1, dually to the first part of the previous case.
Suppose now that $P \xrightarrow{a} \Delta$. The only τ-labelled transition from $C^{l,L}_{<t,e>}[P]$ leads to the probability distribution $\langle \Delta \rangle_l \,|\, \overline{[\![< t', e' >]\!]^l_L}$, where $S \in \lceil \langle \Delta \rangle_l \,|\, \overline{[\![< t', e' >]\!]^l_L} \rceil$ if and only if $S = \langle P' \rangle_l \,|\, [\![< t', e' >]\!]^l_L = C^{l,L}_{<t',e'>}[P']$ and $P' \in \lceil \Delta \rceil$. Therefore,

$$\begin{aligned}
&\mathbf{S}(C^{l,L}_{<t,e>}[P]) = \\
&\quad = \sum_{<P_1,...,P_n> \in\, \omega path(C^{l,L}_{<t,e>}[P])} \textstyle\prod_{i=1}^{n-1} \Delta_i(P_{i+1}) \\
&\quad = \sum_{\{<P_2,...,P_n> |\, <P_1,...,P_n> \in\, \omega path(C^{l,L}_{<t,e>}[P])\}} \Delta(P_2) \cdot \textstyle\prod_{i=2}^{n-1} \Delta_i(P_{i+1}) \\
&\quad = \sum_{\{P' \in \lceil \Delta \rceil \,|\, \omega path(C^{l,L}_{<t',e'>}[P']) \neq \emptyset\}} \Delta(P') \cdot \textstyle\sum_{<P_1,...,P_n> \in\, \omega path(C^{l,L}_{<t',e'>}[P'])} \prod_{i=1}^{n-1} \Delta_{P_i}(P_{i+1}) \\
&\quad = \sum_{\{P' \in \lceil \Delta \rceil \,|\, \omega path(C^{l,L}_{<t',e'>}[P']) \neq \emptyset\}} \Delta(P') \cdot \mathbf{S}(C^{l,L}_{<t',e'>}[P']) \\
&\quad = \sum_{\{P' \in \lceil \Delta \rceil \,|\, \omega path(C^{l,L}_{<t',e'>}[P']) \neq \emptyset\}} \Delta(P') \cdot \mathcal{P}_{t',P'}(e') \qquad \text{(by HI).}
\end{aligned}$$

For any P', $\omega path(C^{l,L}_{<t',e'>}[P']) = \emptyset$ if and only if $\mathbf{S}(C^{l,L}_{<t',e'>}[P']) = 0$ if and only if (by the inductive hypothesis) $\mathcal{P}_{t',P'}(e') = 0$. Therefore,

$$\begin{aligned}
&\sum_{\{P' \in \lceil \Delta \rceil \,|\, \omega path(C^{l,L}_{<t',e'>}[P']) \neq \emptyset\}} \Delta(P') \cdot \mathcal{P}_{t',P'}(e') \\
&= \sum_{P' \in \lceil \Delta \rceil} \Delta(P') \cdot \mathcal{P}_{t',P'}(e') \\
&= \mathcal{P}_{a.t',P}(1_a : e').
\end{aligned}$$

$(< t, e >=< (t_1, ..., t_n), (e_1, ..., e_n) >)$ By induction on n.

If $n = 2$ then we have that:

$$\mathbf{S}(C^{l,L}_{<t',e'>}[P']) = \mathbf{S}(C^{l_1,L_1}_{<t_1,e_1>}[P]\{^{l_1}/_{\omega}\} \,|\, \bar{l}_1.(C^{l_2,L_2}_{<t_2,e_2>}[P]\{^{l_2}/_{\omega}\}) \,|\, \bar{l}_2.\omega \,|\, \mathbf{0})$$

Let $A = C^{l_1,L_1}_{<t_1,e_1>}[P]\{^{l_1}/_{\omega}\} \,|\, \bar{l}_1.(C^{l_2,L_2}_{<t_2,e_2>}[P]\{^{l_2}/_{\omega}\}) \,|\, \bar{l}_2.\omega \,|\, \mathbf{0}$. By using Theorem 3.15 it is easy to check that $< S_1, ..., S_n >\in \omega path(A)$ if and only if there are $m, k \in \mathbb{N}$ such that $n = m + k + 1$ and:

- for all $1 \leq i \leq m$, $S_i = S'_i\{^{l_1}/_{\omega}\} \,|\, \bar{l}_1.(C^{l_2,L_2}_{<t_2,e_2>}[P]\{^{l_2}/_{\omega}\}) \,|\, \bar{l}_2.\omega \,|\, \mathbf{0}$ and $< S'_1, ..., S'_m >\in \omega path(C^{l_1,L_1}_{<t_1,e_1>}[P])$;
- for all $m + 1 \leq i \leq m + k$, $S_i = S' \,|\, S''_i\{^{l_2}/_{\omega}\} \,|\, \bar{l}_2.\omega \,|\, \mathbf{0}$, where $S'_m\{^{l_1}/_{\omega}\} \xrightarrow{l_1} S'$ and $< S''_{m+1}, ..., S''_{m+k} >\in \omega path(C^{l_2,L_2}_{<t_2,e_2>})$;
- $S_{m+k+1} = S' \,|\, S'' \,|\, \omega$, where $S''_{m+k} \xrightarrow{l_2} S''$.

Hence,

$$\begin{aligned}
&\mathbf{S}(A) = \\
&= \sum_{<P_1,...,P_n>\in\, \omega path(A)} \textstyle\prod_{i=1}^{n-1} \Delta_i(P_{i+1}) \\
&= \sum_{<S'_1,...,S'_m>\in \omega path(C^{l_1,L_1}_{<t_1,e_1>}[P])} \Big(\textstyle\prod_{i=1}^{m-1} \Delta_i(S'_{i+1}) \cdot \overline{S_{m+1}}(S_{m+1}) \cdot \sum_{<S''_1,...,S''_k>\in \omega path(C^{l_2,L_2}_{<t_2,e_2>}[P])} \prod_{i=1}^{k-1} \Delta_i(S''_{i+1}) \Big) \\
&= \sum_{<S'_1,...,S'_m>\in \omega path(C^{l_1,L_1}_{<t_1,e_1>}[P])} \textstyle\prod_{i=1}^{m-1} \Delta_i(S'_{i+1}) \quad \cdot \quad \displaystyle\sum_{<S''_1,...,S''_k>\in \omega path(C^{l_2,L_2}_{<t_2,e_2>}[P])} \textstyle\prod_{i=1}^{k-1} \Delta_i(S''_{i+1}) \\
&= \mathcal{P}_{t_1,P}(e_1) \cdot \mathcal{P}_{t_2,P}(e_2) \qquad \text{(by HI)} \\
&= \mathcal{P}_{(t_1,t_2),P}((e_1, e_2))
\end{aligned}$$

The result follows analogously for the inductive case on n.

□

Corollary 3.17. *For all* LS*-processes* P, Q*,* $P \simeq_{\text{may}}^{\mathsf{HOLp}} Q$ *implies* $P \sim Q$ *and* $P \simeq_{\text{must}}^{\mathsf{HOLp}} Q$ *implies* $P \sim Q$*.*

Proof. By Theorem 2.18, if $P \not\sim Q$ then there are a test $t \in \mathbf{T}^{\mathsf{LS}}$ and an observation $e \in O_t$ such that $\mathcal{P}_{t,P}(e) \neq \mathcal{P}_{t,Q}(e)$. It follows from Theorem 3.15 and Theorem 3.16 that:

$$\begin{aligned} & \bigsqcup \mathbb{S}(C^{l,L}_{<t,e>}[P]) = \bigsqcap \mathbb{S}(C^{l,L}_{<t,e>}[P]) = \{\mathcal{P}_{t,P}(e)\} \\ \text{and} \quad & \bigsqcup \mathbb{S}(C^{l,L}_{<t,e>}[Q]) = \bigsqcap \mathbb{S}(C^{l,L}_{<t,e>}[Q]) = \{\mathcal{P}_{t,Q}(e)\}. \end{aligned}$$

Therefore, $P \not\simeq_{\text{may}}^{\mathsf{HOLp}} Q$ and $P \not\simeq_{\text{must}}^{\mathsf{HOLp}} Q$. □

3.4.3 Probabilistic bisimilarity implies *test*-equivalence

This section is devoted to proving that probabilistically bisimilar LS-processes are testing equivalent. To this end, we first show that the processes $C[P]$ and $C[Q]$ obtained by filling the empty hole of a unary HOL-context C with two probabilistically bisimilar LS-processes P, Q are probabilistically bisimilar as well, with respect to the set actions $\mathscr{A}^*_{\tau,\omega}$ (Theorem 3.18). As a consequence, proving that the sets of probabilities of success of probabilistically bisimilar HOLp-processes have the same suprema and infima (Corollary 3.28) turns out to be sufficient for our purposes.

Substitutivity of first-order bisimilarity

Theorem 3.18. *For any* $n \in \mathbb{N}$*, let* $P_1, ..., P_n, Q_1, ..., Q_n$ *be* LS*-processes such that* $P_i \sim Q_i$*, for* $1 \leq i \leq n$*. For every* n*-ary* HOL*-context* C*,* $C[P_1, ..., P_n] \sim C[Q_1, ..., Q_n]$*, where* $\sim$ *is defined on* $\mathscr{A}^*_{\tau,\omega}$*.*

Proof. Consider the following relation on HOLp-processes:

$$\begin{aligned} \mathcal{R} = \{ & < C[P_1, ..., P_n], C[Q_1, ..., Q_n] > \mid n \in \mathbb{N}, C \text{ is an } n\text{-ary } \mathsf{HOL}\text{-context}, \\ & \{P_1, ..., P_n, Q_1, ..., Q_n\} \subseteq Pr(\mathsf{LS}) \text{ and } P_i \sim Q_i \text{ for } 1 \leq i \leq n\}. \end{aligned}$$

In order to show that $\mathcal{R}$ is a probabilistic bisimulation, we prove by structural induction on C that for every $n \in \mathbb{N}$, if C is an n-ary HOL-context and $C[P_1, ..., P_n] \,\mathcal{R}\, C[Q_1, ..., Q_n]$ then:

1. for every $\mu \in \mathscr{A}^*_{\tau,\omega}$ and for every $\Delta \in \mathcal{D}(Pr(\mathsf{HOLp}))$, if $C[P_1, ..., P_n] \xrightarrow{\mu} \Delta$ then $C[Q_1, ..., Q_n] \xrightarrow{\mu} \Theta$ and $\Delta \overline{\mathcal{R}} \Theta$, for some distribution $\Theta \in \mathcal{D}(Pr(\mathsf{HOLp}))$,
2. the symmetric condition.

($C = [\cdot]$) C is a unary context and $C[P] = P \sim Q = C[Q]$. If $P_1 \xrightarrow{\mu} \Delta$ then there is a Θ such that $Q \xrightarrow{\mu} \Theta$ and $\Delta \overline{\sim} \Theta$. Both Δ and Θ are probability distributions on LS-processes and an LS-process $P' \in \lceil \Delta \rceil$ is equal to the substitution of P' in the context $[\cdot]$. Hence, $\Delta \overline{\mathcal{R}} \Theta$. Bisimilarity is an equivalence relation, so the symmetric condition holds as well.

($C = \omega.C'$), ($C = \alpha^*.C'$) The result follows directly from the inductive hypothesis on C'.

($C = \widetilde{\alpha}^*.C'$), ($C = \mathtt{pass}(x)_l.C'$) The processes cannot perform transitions labelled with actions in $\mathscr{A}^*_{\tau,\omega}$, hence they vacuously satisfy the two conditions.

($C = \langle C' \rangle_l$) $\langle C' \rangle_l[P_1, ..., P_n] = \langle C'[P_1, ..., P_n] \rangle_l$ and there are two cases:

- $C'[P_1, ..., P_n] \xrightarrow{\alpha} \Delta$, so by rule (kell) we have that $\langle C'[P_1, ..., P_n] \rangle_l \xrightarrow{\alpha^l} \langle \Delta \rangle_l$. It follows from the inductive hypothesis that there is a probability distribution Θ such that $\langle C'[Q_1, ..., Q_n] \rangle_l \xrightarrow{\alpha^l} \langle \Theta \rangle_l$, where $\Delta \overline{\mathcal{R}} \Theta$. By the definition of $\mathcal{R}$ we have that $\Delta = \sum_{k=1}^{y} p_k \cdot \overline{S_k}$ and $\Theta = \sum_{k=1}^{y} p_k \cdot \overline{S'_k}$, where $S_k \, \mathcal{R} \, S'_k$ for all k, i.e. there is an $m \in \mathbb{N}$ and there are LS-processes $P'_1, ..., P'_m, Q'_1, ..., Q'_m$ such that $S_k = C_k[P'_1, ..., P'_m]$, $S'_k = C_k[Q'_1, ..., Q'_m]$ and $P'_i \, \mathcal{R} \, Q'_i$, for every i from 1 to m. Therefore, $\langle \Delta \rangle_l = \sum_{k=1}^{y} p_k \cdot \overline{\langle C_k[P'_1, ..., P'_m] \rangle_l}$, $\langle \Theta \rangle_l = \sum_{k=1}^{y} p_k \cdot \overline{\langle C_k[Q'_1, ..., Q'_m] \rangle_l}$ and

$$\langle C_k[P'_1, ..., P'_m] \rangle_l = \langle C_k \rangle_l[P'_1, ..., P'_m] \;\mathcal{R}\; \langle C_k \rangle_l[Q'_1, ..., Q'_m] = \langle C_k[Q'_1, ..., Q'_m] \rangle_l.$$

- $C'[P_1, ..., P_n] \xrightarrow{\tau} \Delta$ and $\langle C'[P_1, ..., P_n] \rangle_l \xrightarrow{\tau} \langle \Delta \rangle_l$, by rule ($\tau$kell). The proof is analogous to the previous one.

($C = C_1 \,|\, C_2$) $C[P_1, ..., P_n] = C_1[P_1, ..., P_k] \,|\, C_2[P_{k+1}, ..., P_n]$ and there are five possible cases:

- $C_1[P_1, ..., P_k] \xrightarrow{\mu} \Delta$ and we derive that $C[P_1, ..., P_n] \xrightarrow{\mu} \Delta \,|\, \overline{C_2[P_{k+1}, ..., P_n]}$, by rule (parL).
  By the inductive hypothesis there is a Θ such that $C_1[Q_1, ..., Q_k] \xrightarrow{\mu} \Theta$ and $\Delta \overline{\mathcal{R}} \Theta$; by (parL) we derive that $C_1[Q_1, ..., Q_k] \,|\, C_2[Q_{k+1}, ..., Q_n] \xrightarrow{\mu} \Theta \,|\, \overline{C_2[Q_{k+1}, ..., Q_n]}$. The result follows analogously to the case ($C = \langle C' \rangle_l$).
- $C_2[P_{k+1}, ..., P_n] \xrightarrow{\mu} \Delta$ and, by rule (parR), $C[P_1, ..., P_n] \xrightarrow{\mu} \overline{C_1[P_1, ..., P_k]} \,|\, \Delta$. Symmetrical to the previous case.
- $C_1[P_1, ..., P_k] \xrightarrow{\alpha^*} \Delta_1$, $C_2[P_{k+1}, ..., P_n] \xrightarrow{\bar{\alpha}^*} \Delta_2$ and $C[P_1, ..., P_n] \xrightarrow{\tau} \Delta_1 \,|\, \Delta_2$. By using the inductive hypothesis we can conclude that there are two finite index-sets I, J such that:

- $\Delta_1 \mid \Delta_2 = \sum_{<i,j>} p_{1_i} \cdot p_{2_j} \cdot \overline{C_{1_i}[P'_1, ..., P'_{m_1}] \mid C_{2_j}[P''_1, ..., P''_{m_2}]}$,
- $C[Q_1, ..., Q_n] \xrightarrow{\tau} \Theta_1 \mid \Theta_2$, where:

$$\Theta_1 \mid \Theta_2 = \sum_{<i,j>} p_{1_i} \cdot p_{2_j} \cdot \overline{C_{1_i}[Q'_1, ..., Q'_{m_1}] \mid C_{2_j}[Q''_1, ..., Q''_{m_2}]},$$

- for every pair $< i, j >$,

$$\begin{aligned} &C_{1_i}[P'_1, ..., P'_{m_1}] \mid C_{2_j}[P''_1, ..., P''_{m_2}] = \\ &= C_{1_i} \mid C_{2_j}[P'_1, ..., P'_{m_1}, P''_1, ..., P''_{m_2}] \ \mathcal{R} \ C_{1_i} \mid C_{2_j}[Q'_1, ..., Q'_{m_1}, Q''_1, ..., Q''_{m_2}] = \\ &= C_{1_i}[Q'_1, ..., Q'_{m_1}] \mid C_{2_j}[Q''_1, ..., Q''_{m_2}]. \end{aligned}$$

Therefore, $\Delta_1 \mid \Delta_2 \overline{\mathcal{R}} \Theta_1 \mid \Theta_2$.

- $C_1[P_1, ..., P_k] \xrightarrow{\mathtt{pass}(x)_l} \Delta_1$, $C_2[P_{k+1}, ..., P_n] \xrightarrow{\mathtt{pass}(S)_l} \Delta_2$ and by rule (psync) we derive that $C[P_1, ..., P_n] \xrightarrow{\tau} \Delta_1 \mid \Delta_2\{^S\!/_x\}$.
  $P_1, ..., P_n$ cannot perform higher-order actions, hence $C_1[P_1, ..., P_k] \xrightarrow{\mathtt{pass}(x)_l}$ implies that C_1 is the parallel composition of contexts such that at least one of them is of the form $\mathtt{pass}(x)_l.C'_1$. Analogously, if $C_2[P_{k+1}, ..., P_n] \xrightarrow{\mathtt{pass}(S)_l}$ then C_2 is the parallel composition of contexts such that at least one of them is of the form $\langle C'_2 \rangle$. For the sake of simplicity, suppose that $C_1[P_1, ..., P_k] = \mathtt{pass}(x)_l.C'_1[P_1, ..., P_k]$ and $C_2[P_{k+1}, ..., P_{P_n}] = \langle C'_2[P_{k+1}, ..., P_n] \rangle_l$. The probability distributions Δ_1 and Δ_2 must be $\overline{C'_1[P_1, ..., P_k]}$ and $\overline{\mathbf{0}}$, respectively. Therefore, the distribution reached by $C[P_1, ..., P_n]$ is $\overline{C''_1[P_1, ..., P_k]\{^{C'_2[P_{k+1}, ..., P_n]}\!/_x\} \mid \mathbf{0}}$. The variable x does not appear in $P_1, ..., P_n$, so $C''_1[P_1, ..., P_k]\{^{C'_2[P_{k+1}, ..., P_n]}\!/_x\} = C''_1\{^{C'_2[P_{k+1}, ..., P_n]}\!/_x\}[P_1, ..., P_k]$. Suppose that x occurs free h times in C'_1 and let $e = h \cdot (n - k) + k$. Thus, $C'_1\{^{C'_2}\!/_x\}$ is an e-ary context and we have that $C'_1\{^{C'_2[P_{k+1}, ..., P_n]}\!/_x\} \mid \mathbf{0}[P_1, ..., P_k] = C'_1\{^{C'_2}\!/_x\} \mid \mathbf{0}[P'_1, ..., P'_e]$, where $P'_1, ..., P'_e$ is a sequence of LS-processes in $\{P_1, ..., P_n\}$ preserving the previous substitutions.
  Symmetrically, the process $C[Q_1, ..., Q_n]$ performs a τ-labelled transition to the probability distribution $\overline{C''_1\{^{C'_2[Q_{k+1}, ..., Q_n]}\!/_x\}[Q_1, ..., Q_k] \mid \mathbf{0}}$, which we can rewrite as $\overline{C'_1\{^{C'_2}\!/_x\} \mid \mathbf{0}[Q'_1, ..., Q'_e]}$, where $P'_i \sim Q'_i$ for all i such that $1 \leq i \leq e$. Therefore,

$$\overline{C'_1\{^{C'_2}\!/_x\} \mid \mathbf{0}[P'_1, ..., P'_e]} \ \overline{\mathcal{R}} \ \overline{C'_1\{^{C'_2}\!/_x\} \mid \mathbf{0}[Q'_1, ..., Q'_e]}.$$

- $C_1[P_1, ..., P_k] \xrightarrow{\widetilde{\alpha}\rangle l} \Delta_1$, $C_2[P_{k+1}, ..., P_n] \xrightarrow{\overline{\widetilde{\alpha}}\rangle l} \Delta_2$ and by rule (sync) we derive that $C[P_1, ..., P_n] \xrightarrow{\tau} \Delta_1 \mid \Delta_2$.
  The proof is similar to the one of the previous case: $C_1[P_1, ..., P_k] \xrightarrow{\widetilde{a}\rangle l}$ implies that C_1 is the parallel composition of contexts such that at least one of them is of the form $\widetilde{a}\rangle l.C'_1$, while C_2 must be a parallel composition with a context of the form

$\langle C_2' \rangle$ as a component.

□

As mentioned above, it remains to prove that:

(*) whenever P and Q are probabilistically bisimilar (with respect to the set of actions $\mathscr{A}^*_{\tau,\omega}$) HOLp-processes then $\bigsqcup \mathbb{S}(P) = \bigsqcup \mathbb{S}(P)$ and $\bigsqcap \mathbb{S}(P) = \bigsqcap \mathbb{S}(Q)$.

A first source of difficulty lies in the possibility that HOLp-processes exhibit a divergent behavior.

Example 3.19. Consider the following processes:

$$\begin{aligned} V &= \mathtt{pass}(x)_l.(x \mid \langle x \rangle_l) \\ W &= \bar{a}.\bar{b}.\omega \mid P \\ Z &= V \mid \langle V \mid W \rangle_l \end{aligned}$$

where P is the LS-process such that $P \xrightarrow{a} (\frac{1}{2} \cdot \overline{P} + \frac{1}{2} \cdot \overline{P'})$ and $P' \xrightarrow{b} \overline{P'}$. We have

$$\begin{aligned} Z \xrightarrow{\tau} & \ \overline{V \mid W \mid \langle V \mid W \rangle_l \mid \mathbf{0}} \\ & V \mid W \mid \langle V \mid W \rangle_l \mid \mathbf{0} \xrightarrow{\tau} (\frac{1}{2} \cdot \overline{V \mid \bar{b}.\omega \mid P \mid \langle V \mid W \rangle_l \mid \mathbf{0}} + \frac{1}{2} \cdot \overline{V \mid \bar{b}.\omega \mid P' \mid \langle V \mid W \rangle_l \mid \mathbf{0}}). \end{aligned}$$

where

$$\begin{aligned} V \mid \bar{b}.\omega \mid P' \mid \langle V \mid W \rangle_l \mid \mathbf{0} \xrightarrow{\tau} & \overline{V \mid \omega \mid P' \mid \langle V \mid W \rangle_l \mid \mathbf{0}} \\ & V \mid \omega \mid P' \mid \langle V \mid W \rangle_l \mid \mathbf{0} \xrightarrow{\omega} . \end{aligned}$$

The process $V \mid \bar{b}.\omega \mid P \mid \langle V \mid W \rangle_l \mid \mathbf{0}$ is bisimilar to $\bar{b}.\omega \mid P \mid Z$, thus the sequence of transitions depicted so far can be repeated infinitely many times. Therefore, there exists an oracle $\mathscr{O}$ (i.e. the oracle which always chooses to repeat this sequence) such that there is not a maximal length of the successful paths $\omega path(\mathscr{O}, Z)$ from Z.

Example 3.19 shows that we cannot prove (*) by induction on the maximal length of the successful paths from a process with respect to a given oracle. Moreover, the recursive behavior of Z suggests that a proof by induction on contexts would not be possible as well.

We thereby take a different approach.[39]

[39] The fixed-point approach presented in the following pages exploits some ideas developed in (Deng et al. 2009). In the cited work, the authors deal with finitary probabilistic LTSs, that is, probabilistic LTSs whose set of states is finite but admitting loops. In (Deng et al. 2007) and (Deng et al. 2008), loop-free and finite-state probabilistic LTSs are analyzed.

A fixed-point approach

For the sake of simplicity, we will henceforth assume that the probabilistic LTSs for the processes we consider are trees; in fact, it is easy to check that both bisimilarity and the testing equivalences on HOLp-processes are preserved by unfolding.

Let $Tree(Pr(\mathsf{HOLp}))$ be the set of nodes (or states) in a forest of trees (representing the unfolded LTS for HOLp). The advantage of using tree-like processes is that an oracle on $Tree(Pr(\mathsf{HOLp}))$ can be easily defined as a function assigning to every state P such that $P \xrightarrow{\tau}$ a distribution Δ such that $P \xrightarrow{\tau} \Delta$. In order to satisfy the history-dependence of the schedulers, the oracles are no longer required to take into account the different paths that can lead to a process. This feature is essential for the theory we are going to introduce.

The set $[0,1]^{Tree(Pr(\mathsf{HOLp}))}$ of functions from states in $Tree(Pr(\mathsf{HOLp}))$ to $[0,1]$ equipped with the usual order $\leq$ on functions is a complete lattice.[40] The bottom $\bot$ and top $\top$ elements of the complete lattice are the functions assigning 0 and 1 to every state, respectively. For every oracle $\mathscr{O}$, let the functional $\mathscr{F}_{\mathscr{O}} : [0,1]^{Tree(Pr(\mathsf{HOLp}))} \to [0,1]^{Tree(Pr(\mathsf{HOLp}))}$ be defined as follows:

$$\mathscr{F}_{\mathscr{O}}(f)(P) = \begin{cases} 1 & \text{if } P \xrightarrow{\omega} \\ 0 & \text{if } P \not\xrightarrow{\omega} \text{ and } P \not\xrightarrow{\tau} \\ \sum_{P' \in \lceil \mathscr{O}(P) \rceil} \mathscr{O}(P)(P') \cdot f(P') & \text{if } P \not\xrightarrow{\omega} \text{ and } P \xrightarrow{\tau}. \end{cases}$$

For any sequence $\{\alpha_i\}_{i \in I}$ in a poset, define:

$$\bigsqcup_i \alpha_i = \bigsqcup \{\alpha_i |\ i \in I\}.$$

Lemma 3.20. *Let S be a set and $\{f_i\}_{i \in \mathbb{N}}$ be a non-decreasing sequence of functions in $[0,1]^S$. For every finite subset S' of S,*

$$\bigsqcup_i \sum_{s \in S'} f_i(s) = \sum_{s \in S'} \bigsqcup_i f_i(s).$$

Proof. Let r be a non-negative real number. For every $s \in S'$ there is an $i_s \in \mathbb{N}$ such that for every $j \geq i$, $\bigsqcup_i f_i(s) - f_j(s) \leq r$. Suppose that $|S'| = n$. Then there is an $i_s \in \mathbb{N}$ such that for every $j \geq i_s$, $\bigsqcup_i f_i(s) - f_j(s) \leq \frac{r}{n}$, for every $s \in S'$. Let $m = \max\{i_s |\ s \in S'\}$. It

[40] In what follows, we will avail ourselves of some definitions and results summarized in Appendix A. We refer the reader to (Sangiorgi 2012a) and to (Davies and Priestley 2002) for a thorough presentation of the theory of ordered sets.

follows that for every $s \in S'$ and for every $j \geq m$, $\bigsqcup_i f_i(s) - f_j(s) \leq \frac{r}{n}$. Therefore,

$$(\sum_{s \in S'} \bigsqcup_i f_i(s)) - \sum_{s \in S'} f_j(s) = \sum_{s \in S'} \bigsqcup_i f_i(s) - f_j(s) \leq r$$

for every $j \geq m$ and the result follows. □

Theorem 3.21. *For every oracle $\mathscr{O}$ on $Tree(Pr(\mathsf{HOLp}))$, the functional $\mathscr{F}_{\mathscr{O}}$ is continuous.*

Proof. Let $\{f_i\}_{i \in \mathbb{N}}$ be a non decreasing sequence of functions in $[0,1]^{Tree(Pr(\mathsf{HOLp}))}$. We prove that for every $P \in Tree(Pr(\mathsf{HOLp}))$,

$$(\bigsqcup_i \mathscr{F}_{\mathscr{O}}(f_i))(P) = \mathscr{F}_{\mathscr{O}}(\bigsqcup_i f_i)(P).$$

If $P \xrightarrow{\omega}$ then $\mathscr{F}_{\mathscr{O}}(f_i)(P) = 1$ for all $i \geq 0$. Hence, $(\bigsqcup_i \mathscr{F}_{\mathscr{O}}(f_i))(P) = \bigsqcup_i \mathscr{F}_{\mathscr{O}}(f_i)(P) = \bigsqcup_i 1 = 1 = \mathscr{F}_{\mathscr{O}}(\bigsqcup_i f_i)(P)$. Similarly, $P \overset{\omega}{\not\rightarrow}$ and $P \overset{\tau}{\not\rightarrow}$ imply that $(\bigsqcup_i \mathscr{F}_{\mathscr{O}}(f_i))(P) = 0 = \mathscr{F}_{\mathscr{O}}(\bigsqcup_i f_i)(P)$.

Suppose now that $P \overset{\omega}{\not\rightarrow}$ and $P \xrightarrow{\tau}$.

$$\begin{aligned} \mathscr{F}_{\mathscr{O}}(\bigsqcup_i f_i)(P) &= \sum_{P' \in \lceil \mathscr{O}(P) \rceil} \mathscr{O}(P)(P') \cdot (\bigsqcup_i f_i)(P') = \\ &= \sum_{P' \in \lceil \mathscr{O}(P) \rceil} \mathscr{O}(P)(P') \cdot \bigsqcup_i f_i(P') = \\ &= \sum_{P' \in \lceil \mathscr{O}(P) \rceil} \bigsqcup_i \mathscr{O}(P)(P') \cdot f_i(P'). \end{aligned}$$

For any i, let $F_i : \lceil \mathscr{O}(P) \rceil \to [0,1]$ be the function such that $F_i(P') = \mathscr{O}(P)(P') \cdot f_i(P')$. If $i \leq j$ then $f_i(P') \leq f_j(P')$, so $\mathscr{O}(P)(P') \cdot f_i(P') \leq \mathscr{O}(P)(P') \cdot f_j(P')$, for any $P' \in \lceil \mathscr{O}(P) \rceil$. Therefore, the sequence of functions $\{F_i\}_{i \in \mathbb{N}}$ is non-decreasing and by Lemma 3.20 we have that:

$$\begin{aligned} \sum_{P' \in \lceil \mathscr{O}(P) \rceil} \bigsqcup_i \mathscr{O}(P)(P') \cdot f_i(P') &= \bigsqcup_i \sum_{P' \in \lceil \mathscr{O}(P) \rceil} \mathscr{O}(P)(P') \cdot f_i(P') = \\ &= \bigsqcup_i \mathscr{F}_{\mathscr{O}}(f_i)(P) = \\ &= (\bigsqcup_i \mathscr{F}_{\mathscr{O}}(f_i))(P). \end{aligned}$$

□

It follows from Theorem 3.21 and from the Continuity Theorem on complete lattices that the functional $\mathscr{F}_{\mathscr{O}}$ has a least fixed point $\mathbb{F}_{\mathscr{O}}$ such that $\mathbb{F}_{\mathscr{O}} = \bigsqcup_{n \geq 0} \mathscr{F}^n_{\mathscr{O}}(\bot)$.

Theorem 3.22. *For every oracle $\mathscr{O}$ on $Tree(Pr(\mathsf{HOLp}))$, $\mathbb{S}^{\mathscr{O}} = \mathbb{F}_{\mathscr{O}}$.*

Proof. We prove that:

(1) $\mathbb{S}^{\mathscr{O}}$ is a fixed point of $\mathscr{F}_{\mathscr{O}}$

(2) $\mathbb{S}^{\mathscr{O}} \leq \mathbb{F}_{\mathscr{O}}$

The function $\mathbb{F}_{\mathscr{O}}$ is the least fixed point of $\mathscr{F}_{\mathscr{O}}$, so (by 1) we have that $\mathscr{F}_{\mathscr{O}} \leq \mathbb{S}^{\mathscr{O}}$. Hence, the result follows from the second point.

(1) If $P \xrightarrow{\omega}$ (or $P \overset{\omega}{\not\rightarrow}$ and $P \overset{\tau}{\not\rightarrow}$) then $\mathscr{F}_{\mathscr{O}}(\mathbb{S}^{\mathscr{O}})(P) = \mathbb{S}^{\mathscr{O}}(P)$. Let $P \overset{\omega}{\not\rightarrow}$ and $P \xrightarrow{\tau}$. Then:

$$\mathscr{F}_{\mathscr{O}}(\mathbb{S}^{\mathscr{O}})(P) = \sum_{P' \in \lceil \mathscr{O}(P) \rceil} \mathscr{O}(P)(P') \cdot \mathbb{S}^{\mathscr{O}}(P').$$

It is easy to check that $\mathbb{S}^{\mathscr{O}}(S) > 0$ if and only if $\omega path(\mathscr{O}, S) \neq \emptyset$, for any process S. Thus,

$$\begin{aligned}\sum_{P' \in \lceil \mathscr{O}(P) \rceil} \mathscr{O}(P)(P') \cdot \mathbb{S}^{\mathscr{O}}(P') &= \sum_{\{P' \in \lceil \mathscr{O}(P) \rceil \mid \omega path(\mathscr{O}, P') \neq \emptyset\}} \mathscr{O}(P)(P') \cdot \mathbb{S}^{\mathscr{O}}(P') = \\ &= \sum_{<P_1,\dots,P_n> \in \omega path(\mathscr{O}, P)} \textstyle\prod_{i=1}^{n-1} \mathscr{O}(P_i)(P_{i+1}) = \\ &= \mathbb{S}^{\mathscr{O}}(P).\end{aligned}$$

(2) Let $A = \{pt_1, ..., pt_k\}$ be a finite subset of $\omega path(\mathscr{O}, P)$ and let $max(A) = max\{n | < P_1, ..., P_n >\in A\}$ be the length of the longest path in A. We prove by complete induction on $max(A)$ that $\sum_{<P_1,\dots,P_n> \in A} \prod_{i=1}^{n-1} \mathscr{O}(P_i)(P_{i+1}) \leq \mathbb{F}_{\mathscr{O}}(P)$, for every P and for every finite subset A of $\omega path(\mathscr{O}, P)$. The result is trivial for $max(A) = 0$ or $max(A) = 1$. If $max(A) = m + 2$, then $P \overset{\omega}{\not\rightarrow}$, $P \xrightarrow{\tau}$ and

$$\sum_{<P_1,\dots,P_n> \in A} \textstyle\prod_{i=1}^{n-1} \mathscr{O}(P_i)(P_{i+1}) = \displaystyle\sum_{P' \in A_2} \mathscr{O}(P)(P') \cdot \textstyle\sum_{<P_1,\dots,P_n> \in A_{P'}} \prod_{i=1}^{n-1} \mathscr{O}(P_i)(P_{i+1})$$

where $A_2 = \{P' | P' = P_2$ for some $< P_1, ..., P_n >\in A\} \subseteq \lceil \mathscr{O}(P) \rceil$ and $A_{P'} = \{< P_1, ..., P_n > |P_1 = P'$ and $< P, P_1, ..., P_n >\in A\}$. By definition, $max(A_{P'}) < max(A)$, hence it follows from the inductive hypothesis that:

$$\begin{aligned}\sum_{P' \in A_2} \mathscr{O}(P)(P') \cdot \textstyle\sum_{<P_1,\dots,P_n> \in A_{P'}} \prod_{i=1}^{n-1} \mathscr{O}(P_i)(P_{i+1}) &\leq \displaystyle\sum_{P' \in A_2} \mathscr{O}(P)(P') \cdot \mathbb{F}_{\mathscr{O}}(P') \\ &\leq \sum_{P' \in \lceil \mathscr{O}(P) \rceil} \mathscr{O}(P)(P') \cdot \mathbb{F}_{\mathscr{O}}(P') \\ &= \mathbb{F}_{\mathscr{O}}(P).\end{aligned}$$

Suppose that $pt_1, pt_2, ..., pt_j, ...$ is an enumeration of the successful computations from P, i.e. $\omega path(P) = \{pt_j\}_{j\in\mathbb{N}}$. For any $pt_j =< P_1, ..., P_n >$, let $\prod(pt_j) = \prod_{i=1}^{n-1} \mathscr{O}(P_i)(P_{i+1})$. The sequence $\{\sum_{j=1}^{k} \prod(pt_j)\}_{k\in\mathbb{N}}$ is increasing and bounded, because $\sum_{j=1}^{k} \prod(pt_j) \leq 1$ for any $k \in \mathbb{N}$. Therefore, the limit of the sequence exists and

$$\mathbb{S}^{\mathscr{O}}(P) = \sum_{j=1}^{\infty} \prod(pt_j) = \lim_{k\to\infty} \sum_{j=1}^{k} \prod(pt_j) = \bigsqcup_{k\geq 1} \sum_{j=1}^{k} \prod(pt_j).$$

We have proved that $\sum_{j=1}^{k} \prod(pt_j) \leq \mathbb{F}^{\mathscr{O}}(P)$ for all $k \in \mathbb{N}$, hence

$$\bigsqcup_{k\geq 1} \sum_{j=1}^{k} \prod(pt_j) \leq \mathbb{F}^{\mathscr{O}}(P).$$

□

Extremal testing

We define the functional $\mathscr{F}_{\text{sup}} : [0,1]^{Tree(Pr(\mathsf{HOLp}))} \to [0,1]^{Tree(Pr(\mathsf{HOLp}))}$ as follows:

$$\mathscr{F}_{\text{sup}}(f)(P) = \begin{cases} 1 & \text{if } P \xrightarrow{\omega} \\ 0 & \text{if } P \overset{\omega}{\not\rightarrow} \text{ and } P \overset{\tau}{\not\rightarrow} \\ \bigsqcup\{\sum_{P'\in\lceil\Delta\rceil} \Delta(P')\cdot f(P') \mid P \xrightarrow{\tau} \Delta\} & \text{if } P \overset{\omega}{\not\rightarrow} \text{ and } P \xrightarrow{\tau}. \end{cases}$$

The functional $\mathscr{F}_{\text{inf}}$ is obtained by replacing $\bigsqcup$ with $\bigsqcap$ in $\mathscr{F}_{\text{sup}}$.

Theorem 3.23. *The functionals $\mathscr{F}_{\text{sup}}$ and $\mathscr{F}_{\text{inf}}$ are continuous.*

Proof. Let $\{f_i\}_{i\in\mathbb{N}}$ be a non decreasing sequence of functions in $[0,1]^{Tree(Pr(\mathsf{HOLp}))}$. We prove that for every $P \in Tree(Pr(\mathsf{HOLp}))$,

(1) $(\bigsqcup_i \mathscr{F}_{\text{sup}}(f_i))(P) = \mathscr{F}_{\text{sup}}(\bigsqcup_i f_i)(P)$

(2) $(\bigsqcup_i \mathscr{F}_{\text{inf}}(f_i))(P) = \mathscr{F}_{\text{inf}}(\bigsqcup_i f_i)(P)$.

If $P \xrightarrow{\omega}$ both (1) and (2) follow in the same way as in Theorem 3.21 and we have that:

$$(\bigsqcup_i \mathscr{F}_{\text{sup}}(f_i))(P) = \mathscr{F}_{\text{sup}}(\bigsqcup_i f_i)(P) = (\bigsqcup_i \mathscr{F}_{\text{inf}}(f_i))(P) = \mathscr{F}_{\text{inf}}(\bigsqcup_i f_i)(P) = 1.$$

Similarly, $P \overset{\omega}{\not\rightarrow}$ and $P \overset{\tau}{\not\rightarrow}$ implies that:

$$(\bigsqcup_i \mathscr{F}_{\text{sup}}(f_i))(P) = \mathscr{F}_{\text{sup}}(\bigsqcup_i f_i)(P) = (\bigsqcup_i \mathscr{F}_{\text{inf}}(f_i))(P) = \mathscr{F}_{\text{inf}}(\bigsqcup_i f_i)(P) = 0.$$

Suppose that $P \overset{\omega}{\not\longrightarrow}$ and $P \xrightarrow{\tau}$.

$$\begin{aligned}
\mathscr{F}_{\sup}(\bigsqcup_i f_i)(P) &= \bigsqcup\{\sum_{P'\in\lceil\Delta\rceil} \Delta(P')\cdot(\bigsqcup_i f_i)(P') \mid P \xrightarrow{\tau} \Delta\} = \\
&= \bigsqcup\{\sum_{P'\in\lceil\Delta\rceil} \Delta(P')\cdot\bigsqcup_i f_i(P') \mid P \xrightarrow{\tau} \Delta\} = \\
&= \bigsqcup\{\sum_{P'\in\lceil\Delta\rceil} \bigsqcup_i \Delta(P')\cdot f_i(P') \mid P \xrightarrow{\tau} \Delta\} = \\
&= \bigsqcup\{\bigsqcup_i \sum_{P'\in\lceil\Delta\rceil} \Delta(P')\cdot f_i(P') \mid P \xrightarrow{\tau} \Delta\} = \qquad \text{(by Lemma 3.20)} \\
&= \bigsqcup_{\{(\Delta,i)\mid P\xrightarrow{\tau}\Delta\}} \sum_{P'\in\lceil\Delta\rceil} \Delta(P')\cdot f_i(P') = \\
&= \bigsqcup_{\{(i,\Delta)\mid P\xrightarrow{\tau}\Delta\}} \sum_{P'\in\lceil\Delta\rceil} \Delta(P')\cdot f_i(P') = \\
&= \bigsqcup_i \bigsqcup\{\sum_{P'\in\lceil\Delta\rceil} \Delta(P')\cdot f_i(P') \mid P \xrightarrow{\tau} \Delta\} = \\
&= \bigsqcup_i \mathscr{F}_{\sup}(f_i)(P) = \\
&= (\bigsqcup_i \mathscr{F}_{\sup}(f_i))(P).
\end{aligned}$$

Consider now the equality (2). As above, we derive that:

$$\mathscr{F}_{\inf}(\bigsqcup_i f_i)(P) = \bigsqcap\{\bigsqcup_i \sum_{P'\in\lceil\Delta\rceil} \Delta(P')\cdot f_i(P') \mid P \xrightarrow{\tau} \Delta\}$$

and $$(\bigsqcup_i \mathscr{F}_{\inf}(f_i))(P) = \bigsqcup_i \bigsqcap\{\sum_{P'\in\lceil\Delta\rceil} \Delta(P')\cdot f_i(P') \mid P \xrightarrow{\tau} \Delta\}$$

but in order to prove (2) we need to interchange a supremum with an infimum, instead of two suprema.

For every Δ such that $P \xrightarrow{\tau} \Delta$ and for every i,

$$\bigsqcap\{\sum_{P'\in\lceil\Delta\rceil} \Delta(P')\cdot f_i(P') \mid P \xrightarrow{\tau} \Delta\} \leq \sum_{P'\in\lceil\Delta\rceil} \Delta(P')\cdot f_i(P') \leq \bigsqcup_i \sum_{P'\in\lceil\Delta\rceil} \Delta(P')\cdot f_i(P').$$

Therefore, for every Δ such that $P \xrightarrow{\tau} \Delta$ it holds that:

$$\bigsqcup_i \bigsqcap\{\sum_{P'\in\lceil\Delta\rceil} \Delta(P')\cdot f_i(P') \mid P \xrightarrow{\tau} \Delta\} \leq \sum_{P'\in\lceil\Delta\rceil} \Delta(P')\cdot f_i(P') \leq \bigsqcup_i \sum_{P'\in\lceil\Delta\rceil} \Delta(P')\cdot f_i(P'),$$

which in turn implies that:

$$\bigsqcup_i \bigsqcap \{ \sum_{P' \in \lceil \Delta \rceil} \Delta(P') \cdot f_i(P') |\; P \xrightarrow{\tau} \Delta \} \leq \bigsqcap \{ \bigsqcup_i \sum_{P' \in \lceil \Delta \rceil} \Delta(P') \cdot f_i(P') |\; P \xrightarrow{\tau} \Delta \}.$$

It remains to prove the other direction of the previous inequality.
The processes we consider are image-finite processes, hence $\{\Delta |\; P \xrightarrow{\tau} \Delta\} = \{\Delta_1, ...\Delta_n\}$, for some $n \in \mathbb{N}$; so there is a $\Delta' \in \{\Delta_1, ...\Delta_n\}$ such that:

$$\begin{aligned} &\bigsqcap \{ \bigsqcup_i \sum_{P' \in \lceil \Delta \rceil} \Delta(P') \cdot f_i(P') |\; P \xrightarrow{\tau} \Delta \} = \\ &\qquad = \min \{ \bigsqcup_i \sum_{P' \in \lceil \Delta \rceil} \Delta(P') \cdot f_i(P') |\; \Delta = \Delta_1 \vee ... \vee \Delta = \Delta_n \} \\ &\qquad = \bigsqcup_i \sum_{P' \in \lceil \Delta' \rceil} \Delta'(P') \cdot f_i(P'). \end{aligned}$$

Let $\Theta_1, \Theta_2, \ldots$ be a denumerable sequence of probability distributions such that for every $i \geq 1$, $\Theta_i \in \{\Delta_1, ...\Delta_n\}$ and

$$\sum_{P' \in \lceil \Theta_i \rceil} \Theta_i(P') \cdot f_i(P') = \min \{ \sum_{P' \in \lceil \Delta \rceil} \Delta(P') \cdot f_i(P') |\; P \xrightarrow{\tau} \Delta \}.$$

It follows from the image-finiteness of P that there exists at least one probability distribution $\Delta'' \in \{\Delta_1, ...\Delta_n\}$ which occurs infinitely many times in the sequence $\{\Theta_i\}_{i \in \mathbb{N}}$. Therefore, for every $k \in \mathbb{N}$ there is a $k' \geq k$ such that:

$$\sum_{P' \in \lceil \Delta'' \rceil} \Delta''(P') \cdot f_{k'}(P') = \sum_{P' \in \lceil \Theta_{k'} \rceil} \Theta_{k'}(P') \cdot f_{k'}(P').$$

The sequence of functions $\{f_i\}_{i \in\, \mathbb{N}}$ is non-decreasing, hence the sequence $\{\sum_{P' \in \lceil \Delta'' \rceil} \Delta''(P') \cdot f_i(P')\}_{i \in \mathbb{N}}$ is non-decreasing as well. It follows from the equality above that for every k there is a $k' \geq k$ such that:

$$\sum_{P' \in \lceil \Delta'' \rceil} \Delta''(P') \cdot f_k(P') \leq \sum_{P' \in \lceil \Theta_{k'} \rceil} \Theta_{k'}(P') \cdot f_{k'}(P') \leq \bigsqcup_i \sum_{P' \in \lceil \Theta_i \rceil} \Theta_i(P') \cdot f_i(P').$$

By applying the definition of least upper bound to the left-hand side of the inequality we obtain that:

$$\bigsqcup_i \sum_{P' \in \lceil \Delta'' \rceil} \Delta''(P') \cdot f_i(P') \leq \bigsqcup_i \sum_{P' \in \lceil \Theta_i \rceil} \Theta_i(P') \cdot f_i(P')$$

and the desired inequality follows:

$$\begin{aligned}
\bigsqcup_i \sum_{P'\in\lceil\Delta'\rceil} \Delta'(P')\cdot f_i(P') &= \min\{\bigsqcup_i \sum_{P'\in\lceil\Delta\rceil} \Delta(P')\cdot f_i(P') |\, P \xrightarrow{\tau} \Delta\}\\
&\leq \bigsqcup_i \sum_{P'\in\lceil\Delta''\rceil} \Delta''(P')\cdot f_i(P')\\
&\leq \bigsqcup_i \sum_{P'\in\lceil\Theta_i\rceil} \Theta_i(P')\cdot f_i(P')\\
&= \bigsqcup_i \min\{ \sum_{P'\in\lceil\Delta\rceil} \Delta(P')\cdot f_i(P') |\, P \xrightarrow{\tau} \Delta\}.
\end{aligned}$$

□

By Theorem 3.23 and by the Continuity Theorem on complete lattices it holds that both $\mathscr{F}_{\text{sup}}$ and $\mathscr{F}_{\text{inf}}$ have least fixed points $\mathbb{F}_{\text{sup}}$ and $\mathbb{F}_{\text{inf}}$ such that $\mathbb{F}_{\text{sup}} = \bigsqcup_{n\geq 0} \mathscr{F}^n_{\text{sup}}(\bot)$ and $\mathbb{F}_{\text{inf}} = \bigsqcup_{n\geq 0} \mathscr{F}^n_{\text{inf}}(\bot)$, respectively.

Theorem 3.24. *For every oracle $\mathscr{O}$, $\mathbb{F}_{\text{inf}} \leq \mathbb{F}_{\mathscr{O}} \leq \mathbb{F}_{\text{sup}}$.*

Proof. By induction on n we prove that $\mathscr{F}^n_{\text{inf}}(\bot) \leq \mathscr{F}^n_{\mathscr{O}}(\bot)$ for all $n \in \mathbb{N}$. The case $n = 0$ is trivial. Let $P \overset{\omega}{\not\longrightarrow}$ and $P \xrightarrow{\tau}$.

$$\begin{aligned}
\mathscr{F}^{n+1}_{\text{inf}}(\bot)(P) &= \bigsqcap\{ \sum_{P'\in\lceil\Delta\rceil} \Delta(P')\cdot \mathscr{F}^n_{\text{inf}}(\bot)(P') |\, P \xrightarrow{\tau} \Delta\}\\
&\leq \sum_{P'\in\lceil\mathscr{O}(P)\rceil} \mathscr{O}(P)(P')\cdot \mathscr{F}^n_{\text{inf}}(\bot)(P')\\
&\leq \sum_{P'\in\lceil\mathscr{O}(P)\rceil} \mathscr{O}(P)(P')\cdot \mathscr{F}^n_{\mathscr{O}}(\bot)(P') \qquad \text{(by HI)}\\
&= \mathscr{F}^{n+1}_{\mathscr{O}}(\bot)(P).
\end{aligned}$$

By the continuity of the functionals $\mathscr{F}_{\text{inf}}$ and $\mathscr{F}_{\mathscr{O}}$ we have that $\mathbb{F}_{\text{inf}} = \bigsqcup_{n\geq 0} \mathscr{F}^n_{\text{inf}}(\bot) \leq \bigsqcup_{n\geq 0} \mathscr{F}^n_{\mathscr{O}}(\bot) = \mathbb{F}_{\mathscr{O}}$.
The proof of $\mathbb{F}_{\mathscr{O}} \leq \mathbb{F}_{\text{sup}}$ is similar. □

Let $\bigsqcup_{\mathscr{O}} \mathbb{F}_{\mathscr{O}}$ (respectively: $\bigsqcap_{\mathscr{O}} \mathbb{F}_{\mathscr{O}}$) denote $\bigsqcap\{\mathbb{F}_{\mathscr{O}} |\ \mathscr{O}$ is an oracle on $Tree(Pr(\mathsf{HOLp}))\}$ (respectively: $\bigsqcap\{\mathbb{F}_{\mathscr{O}} |\ \mathscr{O}$ is an oracle on $Tree(Pr(\mathsf{HOLp}))\}$).

Lemma 3.25. *Let Δ be a probability distribution on $Tree(Pr(\mathsf{HOLp}))$.*

$$\bigsqcup_{\mathscr{O}} \sum_{P\in\lceil\Delta\rceil} \Delta(P)\cdot \mathbb{F}_{\mathscr{O}}(P) = \sum_{P\in\lceil\Delta\rceil} \Delta(P)\cdot \bigsqcup_{\mathscr{O}} \mathbb{F}_{\mathscr{O}}(P).$$

Proof. For every $P \in \lceil \Delta \rceil$ and for every oracle $\mathscr{O}$, $\mathbb{F}_{\mathscr{O}}(P) \leq \bigsqcup_{\mathscr{O}} \mathbb{F}_{\mathscr{O}}(P)$. Therefore, $\Delta(P) \cdot \mathbb{F}_{\mathscr{O}}(P) \leq \Delta(P) \cdot \bigsqcup_{\mathscr{O}} \mathbb{F}_{\mathscr{O}}(P)$, which in turn implies that $\sum_{P \in \lceil \Delta \rceil} \Delta(P) \cdot \mathbb{F}_{\mathscr{O}}(P) \leq \sum_{P' \in \lceil \Delta \rceil} \Delta(P) \cdot \bigsqcup_{\mathscr{O}} \mathbb{F}_{\mathscr{O}}(P)$ for every oracle $\mathscr{O}$. By the definition of supremum,

$$\bigsqcup_{\mathscr{O}} \sum_{P \in \lceil \Delta \rceil} \Delta(P) \cdot \mathbb{F}_{\mathscr{O}}(P) \leq \sum_{P \in \lceil \Delta \rceil} \Delta(P) \cdot \bigsqcup_{\mathscr{O}} \mathbb{F}_{\mathscr{O}}(P).$$

In order to prove the other inequality, we show that:

(1) for every positive real number r there exists an oracle $\mathscr{O}$ such that:

$$\sum_{P \in \lceil \Delta \rceil} \Delta(P) \cdot \bigsqcup_{\mathscr{O}} \mathbb{F}_{\mathscr{O}}(P) - \sum_{P \in \lceil \Delta \rceil} \Delta(P) \cdot \mathbb{F}_{\mathscr{O}}(P) < r,$$

(2) the result follows from (1).

(1) Let $\lceil \Delta \rceil = \{P_1 ..., P_n\}$. For every positive real number r and for every $P_i \in \lceil \Delta \rceil$ there is an oracle $\mathscr{O}_i$ such that $\bigsqcup_{\mathscr{O}} \mathbb{F}_{\mathscr{O}}(P_i) - \mathbb{F}_{\mathscr{O}_i}(P_i) < r$. As a consequence, for every $r > 0$ and for every $P_i \in \lceil \Delta \rceil$ there is an oracle $\mathscr{O}_i$ such that $\bigsqcup_{\mathscr{O}} \mathbb{F}_{\mathscr{O}}(P_i) - \mathbb{F}_{\mathscr{O}_i}(P_i) < \frac{r}{n}$. Then $\Delta(P_i) \cdot (\bigsqcup_{\mathscr{O}} \mathbb{F}_{\mathscr{O}}(P_i) - \mathbb{F}_{\mathscr{O}_i}(P_i)) < \frac{r}{n}$. Let $\mathscr{O}'$ be an oracle such that $\mathbb{F}_{\mathscr{O}_i}(P_i) = \mathbb{F}_{\mathscr{O}'}(P_i)$ for every $P_i \in \lceil \Delta \rceil$. Such an oracle can be obtained by defining $\mathscr{O}'(P) = \mathscr{O}_i(P)$ for every $P_i \in \lceil \Delta \rceil$ and for every P reachable from P_i. Hence, for every $r > 0$ and for every $P_i \in \lceil \Delta \rceil$, $\Delta(P_i) \cdot \bigsqcup_{\mathscr{O}} \mathbb{F}_{\mathscr{O}}(P_i) - \Delta(P_i) \cdot \mathbb{F}_{\mathscr{O}'}(P_i) < \frac{r}{n}$. By adding up these values for every $P_i \in \lceil \Delta \rceil$ we obtain that for every $r > 0$ there is an $\mathscr{O}'$ such that $\sum_{P' \in \Delta} \Delta(P) \cdot \bigsqcup_{\mathscr{O}} \mathbb{F}_{\mathscr{O}}(P) - \sum_{P \in \Delta} \Delta(P) \cdot \mathbb{F}_{\mathscr{O}'}(P) < r$.

(2) Suppose that:

$$(\star) \quad \bigsqcup_{\mathscr{O}} \sum_{P \in \lceil \Delta \rceil} \Delta(P) \cdot \mathbb{F}_{\mathscr{O}}(P) < \sum_{P \in \lceil \Delta \rceil} \Delta(P) \cdot \bigsqcup_{\mathscr{O}} \mathbb{F}_{\mathscr{O}}(P).$$

Then $\sum_{P \in \lceil \Delta \rceil} \Delta(P) \cdot \bigsqcup_{\mathscr{O}} \mathbb{F}_{\mathscr{O}}(P) - \bigsqcup_{\mathscr{O}} \sum_{P \in \lceil \Delta \rceil} \Delta(P) \cdot \mathbb{F}_{\mathscr{O}}(P) > 0$, which implies by (1) that there is an oracle $\mathscr{O}'$ such that:

$$\sum_{P \in \lceil \Delta \rceil} \Delta(P) \cdot \bigsqcup_{\mathscr{O}} \mathbb{F}_{\mathscr{O}}(P) - \sum_{P \in \lceil \Delta \rceil} \Delta(P) \cdot \mathbb{F}_{\mathscr{O}'}(P) < \sum_{P \in \lceil \Delta \rceil} \Delta(P) \cdot \bigsqcup_{\mathscr{O}} \mathbb{F}_{\mathscr{O}}(P) - \bigsqcup_{\mathscr{O}} \sum_{P \in \lceil \Delta \rceil} \Delta(P) \cdot \mathbb{F}_{\mathscr{O}}(P).$$

Therefore, $\bigsqcup_{\mathscr{O}} \sum_{P \in \lceil \Delta \rceil} \Delta(P) \cdot \mathbb{F}_{\mathscr{O}}(P) < \sum_{P \in \lceil \Delta \rceil} \Delta(P) \cdot \mathbb{F}_{\mathscr{O}'}(P)$, which contradicts the definition of supremum.

□

Theorem 3.26. *The following equalities hold:*

(1) $\mathbb{F}_{\text{sup}} = \bigsqcup_{\mathscr{O}} \mathbb{F}_{\mathscr{O}}$

(1) $\mathbb{F}_{\text{inf}} = \bigsqcap_{\mathscr{O}} \mathbb{F}_{\mathscr{O}}$

Proof. It follows from Theorem 3.24 that $\mathbb{F}_{\text{inf}} \leq \bigsqcap_{\mathscr{O}} \mathbb{F}_{\mathscr{O}}$ and $\bigsqcup_{\mathscr{O}} \mathbb{F}_{\mathscr{O}} \leq \mathbb{F}_{\text{sup}}$. Hence, it remains to prove that the inequalities in the opposite direction hold as well.

(1) We show that $\bigsqcup_{\mathscr{O}} \mathbb{F}_{\mathscr{O}}$ is a fixed point of $\mathscr{F}_{\text{sup}}$, which implies that $\mathbb{F}_{\text{sup}} \leq \bigsqcup_{\mathscr{O}} \mathbb{F}_{\mathscr{O}}$. If $P \xrightarrow{\omega}$ then $\mathbb{F}_{\mathscr{O}}(P) = 1$ for every oracle $\mathscr{O}$. Hence, $\bigsqcup_{\mathscr{O}} \mathbb{F}_{\mathscr{O}}(P) = 1 = \mathscr{F}_{\text{sup}}(\bigsqcup_{\mathscr{O}} \mathbb{F}_{\mathscr{O}})(P)$. Analogously, if $P \stackrel{\omega}{\not\longrightarrow}$ and $P \stackrel{\tau}{\not\longrightarrow}$ then $\bigsqcup_{\mathscr{O}} \mathbb{F}_{\mathscr{O}}(P) = 0 = \mathscr{F}_{\text{sup}}(\bigsqcup_{\mathscr{O}} \mathbb{F}_{\mathscr{O}})(P)$. Suppose now that $P \stackrel{\omega}{\not\longrightarrow}$ and $P \xrightarrow{\tau}$. We have that:

$$\begin{aligned}
\bigsqcup_{\mathscr{O}} \mathbb{F}_{\mathscr{O}}(P) &= \bigsqcup_{\mathscr{O}} \sum_{P' \in \lceil \mathscr{O}(P) \rceil} \mathscr{O}(P)(P') \cdot \mathbb{F}_{\mathscr{O}}(P') = \\
&= \bigsqcup_{\{\Delta \mid P \xrightarrow{\tau} \Delta\}} \bigsqcup_{\mathscr{O}} \sum_{P' \in \lceil \Delta \rceil} \Delta(P') \cdot \mathbb{F}_{\mathscr{O}}(P') \\
&= \bigsqcup_{\{\Delta \mid P \xrightarrow{\tau} \Delta\}} \sum_{P' \in \lceil \Delta \rceil} \Delta(P') \cdot \bigsqcup_{\mathscr{O}} \mathbb{F}_{\mathscr{O}}(P') \qquad \text{(by Lemma 3.25)} \\
&= \mathbb{F}_{\text{sup}}(\bigsqcup_{\mathscr{O}} \mathbb{F}_{\mathscr{O}})(P).
\end{aligned}$$

(2) Let $\mathscr{O}_m$ be an oracle such that whenever $P \stackrel{\omega}{\not\longrightarrow}$ and $P \xrightarrow{\tau}$,

$$\sum_{P' \in \lceil \mathscr{O}_m(P) \rceil} \mathscr{O}_m(P)(P') \cdot \mathbb{F}_{\text{inf}}(P') \leq \sum_{P' \in \lceil \Delta \rceil} \Delta(P') \cdot \mathbb{F}_{\text{inf}}(P')$$

for every P and for every Δ such that $P \xrightarrow{\tau} \Delta$. We prove that $\mathbb{F}_{\text{inf}}$ is a fixed-point of the functional $\mathscr{F}_{\mathscr{O}_m}$.
If $P \xrightarrow{\omega}$ (respectively: $P \stackrel{\omega}{\not\longrightarrow}$ and $P \stackrel{\tau}{\not\longrightarrow}$) then both $\mathbb{F}_{\text{inf}}(P)$ and $\mathscr{F}_{\mathscr{O}_m}(\mathbb{F}_{\text{inf}})(P)$ are 1 (respectively: 0). If $P \stackrel{\omega}{\not\longrightarrow}$ and $P \xrightarrow{\tau}$ then:

$$\begin{aligned}
\mathscr{F}_{\mathscr{O}_m}(\mathbb{F}_{\text{inf}})(P) &= \sum_{P' \in \lceil \mathscr{O}_m(P) \rceil} \mathscr{O}_m(P)(P') \cdot \mathbb{F}_{\text{inf}}(P') = \\
&= \min\{ \sum_{P' \in \lceil \Delta \rceil} \Delta(P') \cdot \mathbb{F}_{\text{inf}}(P') \mid P \xrightarrow{\tau} \Delta \} = \quad \text{(by the definition of } \mathscr{O}_m\text{)} \\
&= \mathbb{F}_{\text{inf}}(P).
\end{aligned}$$

Therefore, the least fixed point $\mathbb{F}_{\mathscr{O}_m}$ of $\mathscr{F}_{\mathscr{O}_m}$ is below $\mathbb{F}_{\text{inf}}$ and the result follows from the fact that $\bigsqcap_{\mathscr{O}} \mathbb{F}_{\mathscr{O}} \leq \mathbb{F}_{\mathscr{O}_m}$.

□

Theorem 3.27. *Let P, Q be states in $Tree(Pr(\mathsf{HOLp}))$. If $P \sim Q$ then $\mathbb{F}_{\text{sup}}(P) = \mathbb{F}_{\text{sup}}(Q)$ and $\mathbb{F}_{\text{inf}}(P) = \mathbb{F}_{\text{inf}}(Q)$.*

Proof. By induction on n we prove that for all P, Q, if $P \sim Q$ then $\mathscr{F}^n_{\text{sup}}(\bot)(P) = \mathscr{F}^n_{\text{sup}}(\bot)(Q)$ and $\mathscr{F}^n_{\text{inf}}(\bot)(P) = \mathscr{F}^n_{\text{inf}}(\bot)(Q)$.
The case $n = 0$ follows from the definition of $\bot$. For the inductive step, if $P \sim Q$ then there are three possible cases:

1. $P \xrightarrow{\omega}$ and $Q \xrightarrow{\omega}$. In this case, every function returns 1.

2. $P \overset{\omega}{\not\longrightarrow}$, $Q \overset{\omega}{\not\longrightarrow}$, $P \overset{\tau}{\not\longrightarrow}$, $Q \overset{\tau}{\not\longrightarrow}$. Symmetrically, every function returns 0.

3. $P \overset{\omega}{\not\longrightarrow}$, $Q \overset{\omega}{\not\longrightarrow}$, $P \xrightarrow{\tau}$, $Q \xrightarrow{\tau}$.

   If $P \sim Q$ and $P \xrightarrow{\tau} \Delta$, then there exists a probability distribution Θ such that $Q \xrightarrow{\tau} \Theta$ and for some index set I:

$$\begin{aligned}\sum_{P' \in \lceil \Delta \rceil} \Delta(P') \cdot \mathscr{F}^n_{\text{sup}}(\bot)(P') &= \sum_{i \in I} p_i \cdot \mathscr{F}^n_{\text{sup}}(\bot)(P_i) = \\ &= \sum_{i \in I} p_i \cdot \mathscr{F}^n_{\text{sup}}(\bot)(Q_i) = \qquad \text{(by HI)} \\ &= \sum_{Q' \in \lceil \Theta \rceil} \Theta(Q') \cdot \mathscr{F}^n_{\text{sup}}(\bot)(Q').\end{aligned}$$

   Therefore,

$$\{ \sum_{P' \in \lceil \Delta \rceil} \Delta(P') \cdot \mathscr{F}^n_{\text{sup}}(\bot)(P') | \, P \xrightarrow{\tau} \Delta \} \subseteq \{ \sum_{Q' \in \lceil \Theta \rceil} \Theta(Q') \cdot \mathscr{F}^n_{\text{sup}}(\bot)(Q') | \, Q \xrightarrow{\tau} \Theta \}.$$

   It follows from the fact that $\sim$ is an equivalence relation that the opposite inclusion holds too. Hence, we have that:

$$\begin{aligned}\mathscr{F}^{n+1}_{\text{sup}}(\bot)(P) &= \bigsqcup \{ \sum_{P' \in \lceil \Delta \rceil} \Delta(P') \cdot \mathscr{F}^n_{\text{sup}}(\bot)(P') | \, P \xrightarrow{\tau} \Delta \} = \\ &= \bigsqcup \{ \sum_{Q' \in \lceil \Theta \rceil} \Theta(Q') \cdot \mathscr{F}^n_{\text{sup}}(\bot)(Q') | \, Q \xrightarrow{\tau} \Theta \} = \\ &= \mathscr{F}^{n+1}_{\text{sup}}(\bot)(Q).\end{aligned}$$

   The proof for $\mathscr{F}^{n+1}_{\text{inf}}(\bot)$ is similar and the result follows from the continuity of the functionals $\mathscr{F}_{\text{sup}}$ and $\mathscr{F}_{\text{inf}}$.

□

Corollary 3.28. *Let P, Q be states in $Tree(Pr(\mathsf{HOLp}))$. If $P \sim Q$ then $\bigsqcup \mathbb{S}(P) = \bigsqcup \mathbb{S}(Q)$ and $\bigsqcap \mathbb{S}(P) = \bigsqcap \mathbb{S}(Q)$.*

Proof. The result follows from Theorem 3.27, Theorem 3.26 and Theorem 3.22. □

3.4.4 The collapse of equivalences

Theorem 3.29 states the main result of Section 3.4: on any class of LS-processes, probabilistic bisimilarity coincides with both *may*-equivalence and *must*-equivalence.
In Section 3.4.2 we proved that if two LS-processes are not probabilistically bisimilar, then there is a HOL-context differentiating the processes with respect to both *may*-equivalence and *must*-equivalence. The proof of this result exploited an encoding of the tests defined in (Larsen and Skou 1991) and presented in the previous chapter, Section 2.3. Finally, in Section 3.4.3 we proved that HOL-contexts do not discriminate too much, that is, the missing direction of Theorem 3.29: probabilistic bisimilarity on LS-processes implies *test*-equivalence.

Theorem 3.29. *On any class $Pr(\mathsf{LS})$ of LS-processes it holds that:*

$$\sim \;=\; \simeq^{\mathsf{HOLp}}_{\text{may}} \;=\; \simeq^{\mathsf{HOLp}}_{\text{must}} \;=\; \simeq^{\mathsf{HOLp}}_{\text{test}} .$$

Proof. Let P, Q be processes in $Pr(\mathsf{LS})$ such that $P \sim Q$ and let C be a context in $Ctx(\mathsf{HOL})$. It follows from Theorem 3.18 that $C[P] \sim C[Q]$. Let P' and Q' denote the unfolding of $C[P]$ and $C[Q]$, respectively. Bisimilarity is preserved by unfolding, hence $P' \sim Q'$ and, by Theorem 3.28, $\bigsqcup \mathbb{S}(P') = \bigsqcup \mathbb{S}(Q')$ and $\bigsqcap \mathbb{S}(P') = \bigsqcap \mathbb{S}(Q')$. It is easy to check that for every HOLp-process S and for every oracle $\mathscr{O}$ on $Pr(\mathsf{HOLp})$ there is an oracle $\mathscr{O}'$ on $Tree(Pr(\mathsf{HOLp}))$ such that $\mathbb{S}^{\mathscr{O}}(S) = \mathbb{S}^{\mathscr{O}'}(S')$ (where S' is the unfolding of S) and conversely. Therefore, $\bigsqcup \mathbb{S}(C[P]) = \bigsqcup \mathbb{S}(C[Q])$ and $\bigsqcap \mathbb{S}(C[P]) = \bigsqcap \mathbb{S}(C[Q])$, which implies that bisimilarity is included in $\simeq^{\mathsf{HOLp}}_{\text{may}}, \simeq^{\mathsf{HOLp}}_{\text{must}}$ and $\simeq^{\mathsf{HOLp}}_{\text{test}}$.
The opposite inclusion holds by Theorem 3.17, so the result follows.

□

Appendix
Complete lattices and fixed-points

A *poset* (partially ordered set) is a pair $(A, \leq)$ where A is a non-empty set and $\leq$ is a partial order on A, i.e. $\leq$ is a binary relation on A satisfying the following properties:

$$
\begin{array}{ll}
a \leq a & \text{(reflexivity)} \\
a \leq b \,\&\, b \leq a \;\text{ implies }\; a = b & \text{(antisymmetry)} \\
a \leq b \,\&\, b \leq c \;\text{ implies }\; a \leq c & \text{(transitivity)}
\end{array}
$$

for all $a, b, c \in A$.

Let $(A, \leq)$ be a poset and let S be a subset of A.

- An element a of A is an *upper bound* of S if $s \leq a$ for every $s \in S$.
- An element a of A is a *lower bound* of S if $a \leq s$ for every $s \in S$.
- The *greatest element* of S (or *maximum* of S) is an upper bound a of S such that $a \in S$, if such an a exists.
- The *least element* of S (or *minimum* of S) is a lower bound a of S such that $a \in S$, if such an a exists.
- The *least upper bound* of S (or *supremum* or *join* of S) is the least element of the set of upper bounds of S.
- The *greatest lower bound* of S (or *infimum* or *meet* of S) is the greatest element of the set of lower bounds of S.

It is easy to check that if the maximum (respectively: the minimum) of a set exists, then it is unique. As a consequence, the same holds for the supremum and the infimum of a set. We let $\bigsqcup S$ denote the supremum of S and we let $\bigsqcap$ denote the infimum of S, if they exist.

Let $(A, \leq)$ be a poset.
$(A, \leq)$ is a *lattice* if it holds that $\bigsqcup\{a, b\} \in A$, for every pair $a, b \in A$.
$(A, \leq)$ is a *complete lattice* if it holds that $\bigsqcup S \in A$, for every subset $S \subseteq A$.

Theorem 1. *If $(A, \leq)$ is a complete lattice then $\bigsqcap S \in A$, for every subset $S \subseteq A$.*

Proof. Let $S' = \{a \in A |\, a \leq s \text{ for every } s \in S\}$. By the definition of complete lattice, the supremum of S' exists. We prove that $\bigsqcup S' = \bigsqcap S$.
If $s \in S$ and $s' \in S'$ then $s' \leq s$, hence $\bigsqcup S' \leq s$. Suppose now that $a \leq s$ for every $s \in S$. Then $a \in S'$ and it follows from the definition of supremum that $a \leq \bigsqcup S'$.

□

Therefore, both the supremum and the infimum of A exist in a complete lattice $(A, \leq)$. The top $(\top)$ and bottom $(\bot)$ elements of A are its supremum and infimum, respectively.

A function $f : A \to A$ on a complete lattice $(A, \leq)$ is *monotonic* if $a \leq b$ implies $f(a) \leq f(b)$, for all $a, b \in A$.
For any sequence $a_0, a_1, \ldots, a_i, \ldots$, let $\bigsqcup_i a_i = \bigsqcup\{a_i\}_i$.
A function $f : A \to A$ on a complete lattice $(A, \leq)$ is *continuous* if $f(\bigsqcup_i a_i) = \bigsqcup_i f(a_i)$ for every non-decreasing sequence $a_0, a_1, \ldots, a_i, \ldots$.

Theorem 2. *Let $f : A \to A$ be a continuous function on a complete lattice $(A, \leq)$. Then f is monotonic.*

Proof. Consider the nondecreasing sequence $a_0, a_1, \ldots, a_i, \ldots$ such that $a_0 = a$ and $a_i = b$ for $i \geq 1$. By the continuity of f we have that $f(b) = f(\bigsqcup_i a_i) = \bigsqcup_i f(a_i)$, which is above $f(a)$ by the definition of supremum. Hence, $f(a) \leq f(b)$.

□

Let $(A, \leq)$ be a poset and let $f : A \to A$. We say that $a \in A$ is a fixed-point of f if $f(a) = a$ and we let $\mathrm{Fix}(f)$ denote the set of fixed-points of f.

Theorem 3 (Fixed-point Theorem)**.** *Let $f : A \to A$ be a monotone function on a complete lattice $(A, \leq)$. Then $(\mathrm{Fix}(f), \leq)$ is a complete lattice.*

Proof. Let $S \subseteq \mathrm{Fix}(f)$. $(A, \leq)$ is a complete lattice, then $\bigsqcup S$ exists in A. We prove that $\bigsqcup S$ is a fixed point of f.
If $s \in S$ then $s \leq \bigsqcup S$ and by the monotonicity of f we have that $f(s) \leq f(\bigsqcup S)$. Since S is a set of fixed points it holds that $s = f(s)$, so $s \leq f(\bigsqcup S)$ for every $s \in S$. Hence, $\bigsqcup S \leq f(\bigsqcup S)$.
Let $S' = \{a \in A |\, f(a) \leq a \text{ and } \bigsqcup S \leq a\}$. By Theorem 1, the infimum of S' exists and it is easy to check that $\bigsqcap S' = \bigsqcup S$. By the definition of S', if $s \in S'$ then $\bigsqcup S \leq s'$. It

follows from the monotonicity of f that $f(\bigsqcup S) \leq f(s')$. For every $s' \in S'$, $f(s') \leq s'$, then $f(\bigsqcup S) \leq f(s') \leq s'$. Therefore, $f(\bigsqcup S) \leq \bigsqcap S' = \bigsqcup S$ and the result follows.

□

As a consequence of Theorem 3, if $f : A \to A$ is a monotone function on a complete lattice $(A, \leq)$ then the bottom of $\mathrm{Fix}(f)$ exists in $\mathrm{Fix}(f)$ and is the least fixed-point of f (i.e. the least of the set of fixed-points of f). We let $\mathrm{lfp}(f)$ denote the least fixed-point of f.

Given a function $f : A \to A$, let $f^n(a)$ denote the n-th iteration of f on a, for any $a \in A$. The function f^n is defined as follows:

$$\begin{aligned} f^0(a) &= a \\ f^{n+1}(a) &= f(f^n(a)). \end{aligned}$$

Theorem 4 (Continuity Theorem). *Let $f : A \to A$ be a continuous function on a complete lattice $(A, \leq)$. It holds that:*

$$\mathrm{lfp}(f) = \bigsqcup_n f^n(\bot).$$

Proof. We prove that $\bigsqcup_n f^n(\bot)$ is a fixed-point of f and that $\bigsqcup_n f^n(\bot) \leq \mathrm{lfp}(f)$, which implies the result since $\mathrm{lfp}(f)$ is the least fixed-point of f.
By Theorem 2, f is monotonic, so it follows from the fact that $\bot \leq a$ for all $a \in A$ that $f^n(\bot) \leq f^{n+1}(\bot)$ for all $n \in \mathbb{N}$. Therefore, $f^0(\bot), f^1(\bot), \ldots$ is a non-decreasing sequence. By the continuity of f we have that $f(\bigsqcup_n f^n(\bot)) = \bigsqcup_n f(f^n(\bot)) = \bigsqcup_n f^{n+1}(\bot)$. The sequence $f^0(\bot), f^1(\bot), \ldots$ is non-decreasing, so $\bigsqcup_n f^{n+1}(\bot) = \bigsqcup_n f^n(\bot)$ and we have that $\bigsqcup_n f^n(\bot)$ is a fixed-point of f.
Let $n \in \mathbb{N}$. By the definition of bottom, $\bot \leq \mathrm{lfp}(f)$ and by iterating the property of monotonicity we obtain that $f^n(\bot) \leq f^n(\mathrm{lfp}(f))$. By iteratively applying the definition of fixed-point to $\mathrm{lfp}(f)$, we conclude that $f^n(\bot) \leq f^n(\mathrm{lfp}(f)) = \mathrm{lfp}(f)$. Therefore, $f^n(\bot) \leq \mathrm{lfp}(f)$ for every $n \in \mathbb{N}$, which implies that $\bigsqcup_n f^n(\bot) \leq \mathrm{lfp}(f)$.

□

Bibliography

Abramsky, S. (1987). “Observation Equivalence as a Testing Equivalence”. In: *Theoretical Computer Science* 53.2-3, pp. 225 –241.

Aceto, L., A. Ingólfsdóttir, and J. Srba (2012). “The Algorithmics of Bisimilarity”. In: *Advanced Topics in Bisimulation and Coinduction.* Ed. by D. Sangiorgi and J. Rutten. Vol. 52. Cambridge Tracts in Theoretical Computer Science. Cambridge: Cambridge University Press. Chap. 3, pp. 100–174.

Aceto, L. et al. (2007). *Reactive Systems: Modelling, Specification and Verification.* Cambridge: Cambridge University Press.

Aczel, P. (1988). *Non-Well-Founded Sets.* Vol. 14. CSLI Lecture Notes. Stanford: CSLI.

Baeten, J. C. M., ed. (1990). *Applications of Process Algebra.* Vol. 17. Cambridge Tracts in Theoretical Computer Science. Cambridge: Cambridge University Press.

Baeten, J. C. M. (2005). “A brief history of process algebra”. In: *Theoretical Computer Science* 335.2-3, pp. 131 –146.

Baier, C. and M. Kwiatkowska (2000). “Domain Equations for Probabilistic Processes”. In: *Mathematical Structures in Computer Science* 10, pp. 665–717.

Bergstra, J. A. and J. W. Klop (1984). “Process Algebra for Synchronous Communication”. In: *Information and Control* 60.1-3, pp. 109 –137.

Bernardo, M., R. De Nicola, and M. Loreti (2013a). “A Uniform Framework for Modeling Nondeterministic, Probabilistic, Stochastic, or Mixed Processes and their Behavioral Equivalences”. In: *Information and Computation* 225.0, pp. 29 –82.

— (2013b). “Relating Strong Behavioral Equivalences for Processes with Nondeterminism and Probabilities”. To appear in Theoretical Computer Science.

Bernardo, M., R. De Nicola, and M. Loreti (2013c). *Revisiting Bisimilarity and its Modal Logic for Nondeterministic and Probabilistic Processes.* Tech. rep. IMT Institute for Advanced Studies Lucca. URL: `http://eprints.imtlucca.it/id/eprint/1553`.

— (2013d). "The Spectrum of Strong Behavioral Equivalences for Nondeterministic and Probabilistic Processes". In: Proceedings 11th International Workshop on *Quantitative Aspects of Programming Languages and Systems,* Rome, 23rd-24th March 2013. Ed. by L. Bortolussi and H. Wiklicky. Vol. 117. Electronic Proceedings in Theoretical Computer Science. Open Publishing Association, pp. 81–96.

Bertsekas, D. P. and J. N. Tsitsiklis (2008). *Introduction to Probability Theory.* Second Edition. Belmont: Athena Scientific.

Blackburn, P., M. de Rijke, and Y. Venema (2001). *Modal Logic.* Vol. 53. Cambridge Tracts in Theoretical Computer Science. Cambridge: Cambridge University Press.

Bloom, B., S. Istrail, and A. R. Meyer (1995). "Bisimulation Can't Be Traced". In: *Journal of the Association for Computing Machinery* 42.1, pp. 232–268.

Davies, B. A. and H. A. Priestley (2002). *Introduction to Lattices and Order.* Second Edition. Cambridge: Cambridge University Press.

De Nicola, R. and M. Hennessy (1984). "Testing Equivalences for Processes". In: *Theoretical Computer Science* 34, pp. 83–133.

DeGroot, M. H. and M. J. Schervish (2012). *Probability and Statistics.* Fourth Edition. Boston: Addison-Wesley.

Deng, Y. and W. Du (2007). "Probabilistic Barbed Congruence". In: *Electronic Notes in Theoretical Computer Science* 190.3. Also appeared in *Proceedings of the 5th Workshop on Quantitative Aspects of Programming Languages*, pp. 185–203.

— (2011). *Logical, Metric, and Algorithmic Characterisations of Probabilistic Bisimulation.* Tech. rep. CMU-CS-11-110. Carnegie Mellon University.

Deng, Y. et al. (2007). "Remarks on Testing Probabilistic Processes". In: *Electronic Notes in Theoretical Computer Science* 172, pp. 359–397.

Deng, Y. et al. (2008). "Characterising Testing Preorders for Finite Probabilistic Processes". In: *Logical Methods in Computer Science* 4.4:4, pp. 1–33.

— (2009). "Testing Finitary Probabilistic Processes". In: *CONCUR 2009 - Concurrency Theory, 20th International Conference, CONCUR 2009, Bologna, Italy, September 1-*

4, 2009. Proceedings. Ed. by M. Bravetti and G. Zavattaro. Vol. 5710. Lecture Notes in Computer Science. Springer, pp. 274–288.

Desharnais, J., A. Edalat, and P. Panangaden (2002). “Bisimulation for Labelled Markov Processes”. In: *Information and Computation* 179, pp. 163–193.

Desharnais, J. et al. (2003). “Approximating Labelled Markov Processes”. In: *Information and Computation* 184, pp. 160–200.

Goldblatt, R. (2006). “Mathematical modal logic: A view of its evolution”. In: *Logic and the Modalities in the Twentieth Century.* Ed. by D. M. Gabbay and J. Woods. Vol. 7. Handbook of the History of Logic. Amsterdam: Elsevier, pp. 1 –98.

Gorrieri, R. (2013). “Introduction to Concurrency Theory”. Manuscript.

Hennessy, M. (2012). “Exploring Probabilistic Bisimulations, Part I”. In: *Formal Aspects of Computing* 24.4-6, pp. 749–768.

Hennessy, M. and R. Milner (1985). “Algebraic Laws for Nondeterminism and Concurrency”. In: *Journal of the Association for Computing Machinery* 32.1, pp. 137–161.

Hermanns, H. et al. (2011). “Probabilistic Logical Characterization”. In: *Information and Computation* 209.2, pp. 154–172.

Hoare, C. A. R. (1978). “Communicating Sequential Processes”. In: *Communications of the ACM* 21.8, pp. 666–677.

— (1980). “Communicating Sequential Processes”. In: *On the Construction of Programs - An Advanced Course.* Ed. by R. M. McKeag and A. M. Macnaghten. New York: Cambridge University Press, pp. 229–254.

Jacobs, B. and J. Rutten (2012). “An Introduction to (co)algebra and (co)induction”. In: *Advanced Topics in Bisimulation and Coinduction.* Ed. by D. Sangiorgi and J. Rutten. Vol. 52. Cambridge Tracts in Theoretical Computer Science. Cambridge: Cambridge University Press. Chap. 2, pp. 38–99.

Kripke, S. (1963). “Semantical Considerations on Modal Logic”. In: *Acta Philosophica Fennica* 16, pp. 83–94.

Larsen, K. G. and A. Skou (1991). “Bisimulation through Probabilistic Testing”. In: *Information and Computation* 94.1, pp. 1–28.

Milner, R. (1971). “An Algebraic Definition of Simulation Between Programs”. In: *Proceeding of the 2nd International Joint Conferences on Artificial Intelligence.* British Computer Society. London.

— (1980). *A Calculus of Communicating Systems.* Ed. by G. Goos and J. Hartmanis. Vol. 92. Lecture Notes in Computer Science. Berlin: Springer.

— (1981). “A Modal Characterisation of Observable Machine-behaviour”. In: *CAAP ’81.* Ed. by E. Astesiano and C. B´ohm. Vol. 112. Lecture Notes in Computer Science. Berlin: Springer, pp. 25–34.

— (1983). “Calculi for Synchrony and Asynchrony”. In: *Theoretical Computer Science* 25, pp. 267–310.

— (1989). *Communication and Concurrency.* Exeter: Prentice Hall.

Park, D. (1981). “Concurrency and Automata on Infinite Sequences”. In: *Conference on Theoretical Computer Science.* Ed. by P. Deussen. Vol. 104. Springer, pp. 167–183.

Parma, A. and R. Segala (2007). “Logical Characterizations of Bisimulations for Discrete Probabilistic Systems”. In: *Foundations of Software Science and Computational Structures.* Ed. by H. Seidl. Vol. 4423. Lecture Notes in Computer Science. Berlin: Springer, pp. 287–301.

Plotkin, G. D. (2004a). “A Structural Approach to Operational Semantics”. In: *The Journal of Logic and Algebraic Programming* 60–61, pp. 17 –139.

— (2004b). “The Origins of Structural Operational Semantics”. In: *The Journal of Logic and Algebraic Programming* 60–61.0, pp. 3 –15.

Puterman, M. L. (1994). *Markov Decision Processes.* Wiley Series in Probability and Mathematical Statistics. Hoboken: John Wiley & Sons.

Sangiorgi, D. (2012a). *Introduction to Bisimulation and Coinduction.* Cambridge: Cambridge University Press.

— (2012b). “On the origins of bisimulation and coinduction”. In: *Advanced Topics in Bisimulation and Coinduction.* Ed. by D. Sangiorgi and J. Rutten. Vol. 52. Cambridge Tracts in Theoretical Computer Science. Cambridge: Cambridge University Press. Chap. 1, pp. 1–37.

Sangiorgi, D. and D. Walker (2001). *The π-calculus: a Theory of Mobile Processes.* Cambridge: Cambridge University Press.

Schmitt, A. and J. Stefani (2005). "The Kell Calculus: A Family of Higher-Order Distributed Process Calculi". In: *Global Computing*. Ed. by C. Priami and P. Quaglia. Vol. 3267. Lecture Notes in Computer Science. Berlin: Springer, pp. 146–178.

Segala, R. and N. Lynch (1994). "Probabilistic simulations for probabilistic processes". In: *CONCUR '94: Concurrency Theory*. Ed. by B. Jonsson and J. Parrow. Vol. 836. Lecture Notes in Computer Science. Berlin: Springer, pp. 481–496.

— (1995). "Probabilistic simulations for probabilistic processes". In: *Nordic Journal of Computing* 2.2, pp. 250–273.

Sokolova, A. and E. P. Vink (2004). "Probabilistic Automata: System Types, Parallel Composition and Comparison". In: *Validation of Stochastic Systems*. Ed. by C. Baier et al. Vol. 2925. Lecture Notes in Computer Science. Berlin: Springer, pp. 1–43.

van Benthem, J. (1983). *Modal Logic and Classical Logic*. Napoli: Bibliopolis.

van Glabbeek, R. J. (2001). "The Linear Time - Branching Time Spectrum I. The Semantics of Concrete, Sequential Processes". In: *Handbook of Process Algebra*. Ed. by J. A. Bergstra, A. Ponse, and S.A. Smolka. Amsterdam: Elsevier, pp. 3–99.

van Glabbeek, R. J., S. A. Smolka, and B. Steffen (1995). "Reactive, Generative and Stratified Models of Probabilistic Processes". In: *Information and Computation Comput.* 121.1, pp. 59–80.

van Glabbeek, R. J. et al. (1990). "Reactive, Generative, and Stratified Models of Probabilistic Processes". In: *LICS. Proceedings of the Fifth Annual Symposium on Logic in Computer Science (LICS '90), Philadelphia, Pennsylvania, USA, June 4-7, 1990.* IEEE Computer Society, pp. 130–141.

Yi, W. and K. G. Larsen (1992). "Testing Probabilistic and Nondeterministic Processes". In: *PSTV-Protocol Specification, Testing and Verification XII, Proceeding of the IFIP TC6/WG6.1 Twelfth International Symposium on Protocol Specification, Testing and Verification*. Ed. by R. J. Linn Jr. and M. Ümit Uyar. Vol. C-8. IFIP Transactions. North-Holland, pp. 47–61.