EQUIVALENCES ON PHASE TYPE PROCESSES

Inauguraldissertation
zur Erlangung des akademischen Grades
eines Doktors der Naturwissenschaften
der Universität Mannheim

vorgelegt von

Verena Wolf
aus Hachenburg

Mannheim, 2008

Dekan:	Prof. Dr. M. Krause, Universität Mannheim
Referentin:	Prof. Dr. M. Majster-Cederbaum, Universität Mannheim
Korreferentin:	Prof. Dr. C. Baier, Technische Universität Dresden

Tag der mündlichen Prüfung: 15. April 2008

Abstract

In this thesis, we introduce Phase Type Processes (PTPs), a novel stochastic modeling approach that can express probabilistic and nondeterministic choices as well as random delays following phase type distributions, a generalization of exponential distributions. Action-labeled transitions are used to react on external stimuli and they are clearly separated from phase type transitions. The semantics of PTPs are defined in terms of path probabilities with respect to schedulers that resolve nondeterministic choices based on the timed process history.

The main emphasis of this work is to analyze a variety of notions of equivalence for PTPs and classify them with respect to their distinguishing power. Amongst others, we define bisimulation, trace and testing equivalence as well as extensions of failure trace equivalence. Moreover, the contribution includes a discussion of parallel composition in the context of a partial memoryless property and the examination of a mapping from PTPs to the subclass of single phased processes in which all random delays are exponentially distributed.

Zusammenfassung

Die vorliegende Arbeit analysiert und diskutiert einen neuartigen Ansatz zur Modellierung stochastischer Systeme, der auf sogenannten *Phasentyp-Prozessen* (PTPs) basiert. PTPs bieten die Möglichkeit von probablistischen Verzweigungen und Zustandsübergängen, die entweder mit einer zufälligen zeitlichen Verzögerung stattfinden oder eine (unverzögerte) atomare Aktion repräsentieren. Die Verzögerung eines zeitbehafteten Übergangs wird durch eine phasentyp-verteilte Zufallsvariable beschrieben. Phasentypverteilungen stellen eine Verallgemeinerung von Exponentialverteilungen dar und besitzen nur eine partielle Gedächtnislosigkeit. Atomare Aktionen ermöglichen eine Interaktion mit der Systemumgebung und ihre Ausführung ist klar getrennt von den Phasentypübergängen. Die Semantik eines PTPs wird durch die Definition eines Wahrscheinlichkeitsraumes über Ausführungssequenzen angegeben. Diese Sequenzen folgen den Regeln eines *Schedulers*, einer Instanz zur Auflösung nichtdeterministischer Entscheidungen basierend auf der Prozessvergangenheit.

Der Schwerpunkt dieser Arbeit liegt auf der Analyse einer Vielzahl von Äquivalenzbegriffen für PTPs, die entsprechend ihrer Unterscheidungsfeinheit klassifiziert werden. Unter anderem werden Bisimulations-, Spur- und Testäquivalenz, sowie Erweiterungen der Spuräquivalenz definiert. Weiterhin beinhaltet die Arbeit eine Diskussion der parallelen Komposition von PTPs und die Untersuchung eines Operators, der PTPs auf Ein-Phasentyp-Prozesse abbildet. Unter Verwendung des Operators und der Tatsache, dass Ein-Phasentyp-Prozesse die volle Gedächtnislosigkeit besitzen, werden mögliche Auswege aus der Problematik der parallelen Komposition im Kontext einer partiellen Gedächtnislosigkeit diskutiert.

Acknowledgments

Foremost, I would like to thank my two supervisors, Prof. Mila Majster-Cederbaum and Prof. Christel Baier, for their guidance and encouragement during my research and study at the university of Mannheim. I count myself lucky to represent the youngest generation of Mannheim's "scientific women in concurrency". My supervisors have always been accessible and willing to help me with my research and were valuable consultants for my scientific development.

I also wish to thank Prof. Holger Hermanns for the numerous useful and enjoyable discussions.

I want to express my appreciation to Werner Sandmann from whom I learned the most in stochastic modeling. He has also been a good adviser and motivated me in difficult times.

Thanks are also due to Daniel Klink for helping with proof reading the manuscript. I also gratefully acknowledge Rita Sommer, the group secretary, and Jürgen Jaap, the group system administrator, and my colleagues in Mannheim (especially Christoph Minnameier, Moritz Martens and Nils Semmelrock).

I express my deep gratitude to my parents and my sister; this dissertation is simply impossible without them and their care and support are of most importance to me.

I would regret my doctoral years in Mannheim if I did not move to Seckenheim where I shared the apartment with Ulrike Edel, Sebastian Biniek, Wolfgang Schmid, Anja Anton and Pablo Krämer.

My very best friends in Mannheim, Monika Huber and Julia Keinert, helped me so much to get through difficult times and provided emotional support

for which I am deeply grateful. Thanks.
I also wish to thank Martina Schneider and Susanne Neumann for so many enjoyable hours at the stable. Many thanks to my horse (Gosik) for not bucking me off whenever I vent my frustrations out on it.
Finally, I would like to mention that I was extraordinarily fortunate in having met Hans whose company and wisdoms I value dearly. Perhaps because of my propensity to philosophy I have learned to defy all those restrictions that life has placed upon me.

Contents

Chapter 1

Introduction

1.1 Motivation

A *mathematical model* is usually a simplification of a real system under study and is used to provide deeper insight into the system behavior. The construction of a model is based on an abstract view on the system hiding unnecessary parts. As remarked by Wolkenhauer and Mesarović [WM05]

> *Mathematical modeling is therefore an art, not unlike writing short stories or aphorisms. A complex story, fact or reality is condensed to few essential aspects.*

Mathematical models are omnipresent in science, engineering, economics, and medicine. Their solution helps to understand the almost always too complicated real world's behavior. For example in the field of systems biology, mathematical modeling is used for the prediction of phenotypic behavior and for the understanding of molecular mechanisms. In manufacturing, performance measures like throughput and utilization of workstations are calculated to identify bottlenecks.

In most cases the model is highly complex because the desired accuracy of the analysis results require the incorporation of many system details. They are susceptible to design errors and in order to ease the model construction high-level modeling frameworks are preferable.

An essential feature of many high-level modeling languages is *compositionality*, i.e. the system can be fractionized into several subsystems and each of these is modeled independently. Finally, the submodels are combined to form the global model. This last step requires sophisticated operators which ensure that the composition of submodels is appropriate.

Besides the advantage of a modular design, compositionality provides the possibility to view the system on different levels of abstraction which is important during the intricate modeling process. Moreover, many modeling languages allow hierarchical modeling which means that one starts with a coarse simplification passing through several refinement steps. A component of the model can be replaced by one which describes the system more detailed.

Compositionality may also allow for a modular analysis, i.e. the solutions of the submodels considered in isolation are combined to a solution of the global model. This technique is exceedingly helpful in the case of models for which standard analysis is not feasible.

Uncertainty and randomness are nearly ubiquitous in the real world and the mathematical study of such phenomena has a wide variety of applications. Often random assumptions make sense and it is necessary to model stochastic phenomena like failures (e.g. of an unreliable communication medium), random time delays, or an unknown environment. It may also be the case that random elements are part of the real system (e.g. randomized algorithms or *stochasticity* in cells [KEBC05]).

The usual approach has been to replace nondeterminism by probabilistic branching. However, in many cases the exact probabilities are unknown and it is more advantageous to model both, probabilities and nondeterminism. This allows for

- underspecification, partly removed in refinement steps, and
- the representation of incomplete information on parameters (such as Milner's "weather conditions" [Mil89]).

Let us consider, for instance, the specification of a communication channel with message loss. Refinement can decrease the set of allowed message loss probabilities.

In the context of compositional stochastic modeling frameworks *stochastic process algebras* (SPAs) have emerged as a useful approach by which systems exhibiting random behavior can be specified and analyzed. The most popular SPAs are IMCs [Her02], EMPA [BG96], PEPA [Hil96], stochastic π-calculus [Pri95] and TIPP [GHR92]. They are all based on the idea that the system evolves from its current state to the next one after a delay which is of exponentially distributed length. The exponential distribution possesses the so-called *memoryless property* which ensures that the underlying stochastic process is a *continuous time Markov process*. In the case of a discrete state space, the process is called a chain and has a simple state transition diagram representation. Both, modeling and analysis, become very much easier if the underlying stochastic process is a *continuous time Markov chain*.

Unfortunately, delays in real-world systems often do not follow exponential distributions, but rather follow distributions "with memory". For example, distributions of CPU service times, file sizes, transfer times, call center service times, channel holding times in cellular networks deviate considerably from exponential (see [MSZ00, PF95, JL96], for instance). On the other hand, many stochastic models become numerically tractable when distributions are assumed to be exponential. This is because of the memoryless property.

Another reason not to use exponential distributions as it is done in SPAs is that one might want to abstract from combinations of exponential delays. For example, in many stochastic models of gene expression it is assumed that each elongation step during gene transcription has exponential duration. The intermediate states, however, represent details of the system, one wants to hide in most cases. More precisely, if one is interested in the quantities of gene expression products and decides that no further details concerning elongation are modeled (such as RNA polymerase elongation factors which regulate the rate of transcription), a more abstract view in which the successive elongation steps are combined to one step is much more appropriate (see, for example, [ARM98]).

A possible solution to these problems is the use of *phase type distributions* (PH distributions) which describe combinations of exponential distributions as a single distribution. This formulation allows the Markov structure of stochastic models to be retained when they replace the familiar exponential distribution. PH distributions are defined as distributions of absorption times in finite Markov chains in which all states are transient except one absorbing state. It has been shown by Johnson and Taaffe that PH distributions are dense in the field of all positive valued distributions, i.e. they can be used to approximate any kind of probability distribution on $[0, \infty)$ [JT88]. PH distributions possess a partial memoryless property since a phase variable can be used to keep track of the state of the underlying Markov process. Phase type models have turned out to be analytically and numerically tractable if the number of parameters is small. However, it is important to keep in mind that there exist distributions for which large numbers of parameters are necessary to give a satisfying approximation [O'C99]. But on the other hand, even if a small number of parameters is used, a large variety of distributions which are of great practical interest can be approximated accurately.

In this thesis, we present a stochastic modeling approach which is based on phase type distributed delays. The resulting state transition diagrams, called *phase type processes*, are equipped with communication actions modeling the system's response to external stimuli. Essentially, a phase type process can be regarded as a state transition graph that combines the features of continuous time Markov chains with those of labeled transition systems. The clear separation between action-based communication structure and internal stochastic behavior is similar to Hermanns' interactive Markov chains [Her02] and so are the probabilistic branching possibilities to those of Segala's probabilistic automata [Seg95].

To the best of our knowledge, in the context of concurrent processes phase type distributions have not yet been incorporated in the way it is done in this thesis. In [HK00] the authors present an elapse operator for interactive Markov chains that describes a phase type distributed delay. Their notion of bisimulation for interactive Markov chains is defined independently from the operator. Therefore, they are not able to abstract from the structure of the Markov chain representing the phase type distribution. Here, we present a bisimulation equivalence for phase type processes that does so. The main difference between the approach in [HK00] and the approach in this thesis is, however, that we give semantics in terms of schedulers that have no knowledge about the current phase of the delay assigned to a transition. It is important to point out that a combination of Markovian transitions in an interactive Markov chain (which does represent a phase type distribution) has a different semantics than a timed transition in a phase type process (representing the same distribution). Moreover, the phase type process explicitly defines the "macro" states of the system that are of interest and hides all "micro" states that belong to a certain phase of the delay in this macro state. Minor differences concern the possibility of probabilistic branching and the fact that we focus on linear time relations here.

El-Rayes et al. present in [ERKN99] an extension of the stochastic process algebra PEPA [Hil96] in which actions have a phase type distributed duration. The same idea is picked up in [Gaj96] but on the basis of the TIPP calculus [GHR92]. El-Rayes et al. make use of matrix geometric methods [Neu81] to solve the underlying infinite-state Markov chain. Their calculus does not allow for nondeterminism and they do not consider any notion of equivalence but rather focus on the derivation of the underlying Markov chain and its solution.

There has also been work done on stochastic process algebras with general distributions (see, for instance, [BD04] and the references therein). The key idea is to use clocks in order to combat the problem of residual lifetimes of delays. In our setting, we do not need this construct because it is enough to record the current phase of a delay.

As opposed to models based on exponential delays or to those based on general distributions, for phase type processes the memoryless assumption is partially valid. This has great impact on most concepts developed in this thesis. Hermanns found an elegant way to circumvent the problems related to the execution of synchronous actions in stochastic process algebras (see [Her02] and Section 4.1 for a detailed discussion). An equivalent approach fails in the more general setting of phase type processes. We present a parallel composition operator on the subclass of *single phase type processes*. By defining a mapping from arbitrary phase type processes to single phased ones, we are able to analyze parallel composition against the background of a partial memoryless property. However, applying this mapping extends the possible behavior of the process and the resolution of occurring nondeterminism must be done with respect to the original process.

Implementation relations, such as bisimulation equivalence [Mil80] or trace equivalence [Hoa80], are central for both, the design of complex systems and the analysis by abstraction. For labeled transition systems, various implementation relations have been suggested (see e.g. [vGla90] for an overview of the most important relations from the linear time - branching time spectrum) and studied under several aspects such as congruence properties with respect to composition operators, axiomatization, algorithms for checking equivalence and logical, domain-theoretic and coalgebraic characterizations.

In the past 15 years, many researchers suggested extensions of the equivalences and preorders originally introduced for labeled transition systems to reason about quantitative aspects such as time or probabilities. Many relations have been studied for models with discrete probabilities (see e.g. [BH97, BKHW05, Her02, LS91, PLS00, SL95] for bisimulation-like relations, [HT92a, Low93, Seg95] for trace and failure semantics, and [Chr90, CSZ92, Seg96, KCS98, JY02, JY95, KN98, SV03] for testing relations), while research on implementation relations for continuous time stochastic models mainly concentrated on the branching time view.

For processes acting in continuous time without the possibility of nondeterministic branching, bisimulation and simulation relations have been studied under various aspects, see, for instance, [BKHW05, BG96, Buc94, Hil96] and the references therein. Testing and trace equivalences have been addressed in [Ber07, BB07, BC00, WMB05]. To the best of our knowledge, the only reference in which linear time relations are analyzed in a framework with continuous time and nondeterminism is [WBM06].

In this thesis, we deal with nondeterministic and probabilistic branching and the processes under study are time-aware, as they have an explicit reference to time. This class of processes is a proper superset of many of those on which the work mentioned above is based. We concentrate on linear time equivalences for phase type processes and classify them according to their distinguishing power. A strong notion of bisimulation equivalence is also defined and used whenever a very low level of abstraction is appropriate, i.e., whenever we want to have much distinguishing power.

Relations for concurrent processes based on observations are often motivated by the fact that bisimulation-like equivalences distinguish processes that are equal on the desired level of abstraction. Lowe, for example, writes in [Low93]:

> *I do not believe that bisimulation is the equivalence that we want, because it makes more distinctions than we would really like.*

For most applications, the point where internal probabilistic branching occurs is not important. As opposed to bisimulation, equivalences based on testing are not sensitive to probabilistic branching. As stated by Kwiatkowska and Norman [KN98]:

> *The idea for this [testing] equivalence is to only make distinctions that are in some sense observable [...].*

Consider the simple time-abstract example of two phase type processes in Figure 1.1. Although phase type processes have not been introduced yet, it

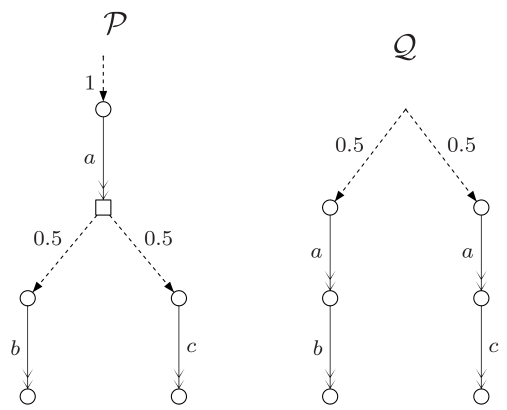

Figure 1.1: $\mathcal{P}$ and $\mathcal{Q}$ are not bisimulation equivalent.

is easy to see that $\mathcal{P}$ and $\mathcal{Q}$ are equivalent in some sense because they show the same observable behavior under all environment conditions although the choice between observation ab and ac is, in the case of $\mathcal{P}$, made after the execution of a and, in the case of $\mathcal{Q}$, before a.

We define trace equivalence for phase type processes based on testing scenarios which are in the style of button pushing experiments as in [vG93, vGla90]. Different notions of trace equivalence arise by varying the way nondeterministic choices are resolved by *schedulers*. Schedulers randomly decide for a possible branch the process can follow. Surprisingly, in most cases there is no correlation between the containment relation of the scheduler classes and the distinguishing power of the induced notions of trace equivalence. We also treat variants of trace semantics, namely semantics based on completed traces [BW82], failures [BHR84] and ready sets [OH83].

We pursue the idea of button pushing experiments and develop testing scenarios in which a process is observed under certain environment conditions. The basic ideas of failure traces [Phi87] and ready traces [Pnu85, BB87] are extended to the continuous time probabilistic setting. Obviously, it makes sense to assume that the environment of a phase type process is time-aware and provides external stimuli after a certain amount of time. Equivalently, one can imagine a scheduler that decides to wait until it makes a choice. Moreover, we obtain as a side effect a notion of schedulers that resolve only internal nondeterminism. The relations resulting from these considerations are analyzed with respect to their relationship to the remaining notions of equivalence. Additionally, we check whether they are compatible with the parallel operator on single phase type processes (*congruence property*). It turns out that neither the relations based on the simulation of a (time-abstract) probabilistic environment nor those based on the simulation of a continuous time stochastic environment are powerful enough to ensure the congruence property. Only the testing equivalence we define for single phase type processes is, by definition, a congruence.

We do not consider algorithms to check whether two processes are related or not (for any of the defined relations) as this exceeds the scope of this thesis. But we give hints for checking bisimulation equivalence.

The main contribution of this thesis is

- the definition and semantics of phase type processes leading to a novel stochastic modeling paradigm,
- the discussion of the problems which arise in the context of parallel composition of Markov models with a partial memoryless property and nondeterminism,
- the definition and comparison of various trace equivalences including a large number of interesting counterexamples which give deeper insight in the distinguishing power of the equivalences,
- the extension of ideas from failure and ready trace equivalence to the continuous time stochastic setting leading to new types of schedulers and time-sensitive relations based on trace observations,
- the classification of all defined relations according to their distinguishing power including a comparison with bisimulation and testing equivalence.

1.2 Road Map

In Chapter 2 some preliminary definitions of operations related to the Kronecker product of matrices are given as well as a short introduction of probability spaces, random variables and their distributions. Moreover, the last two sections constitute a sufficient preparation in continuous time Markov chains and phase type distributions.

Chapter 3 provides a detailed exposition of phase type processes. After the scheduler definition and a classification of schedulers, semantics are given in terms of path probabilities. We proceed with a notion of bisimulation equivalence for phase type processes and establish the link between the equivalence and the path probabilities. Finally, we investigate the use of phase type transitions that are taken immediately with a non-zero probability (which is prohibited in the preceding definition of phase type processes).

We focus on the parallel composition in Chapter 4. In the case of phase type processes that exclusively contain distributions with at most one phase, called single phase type processes, the definition of a composed process is straight forward. In order to be able to discuss the general case, we introduce an "expand" operator that maps a phase type process to a single phase type process. The chapter concludes with a discussion of a parallel composition operator for phase type processes.

In Chapter 5 we present relations based on trace observations and study the influence of the chosen scheduler type on the distinguishing power of the relation.

Chapter 6 deals with more advanced relations based on trace observations. Phase type processes while operating in a time-abstract but probabilistic environment and a time-aware probabilistic environment are analyzed, respectively. This sheds some new light on the way nondeterminism is resolved in phase type processes and relevant counterexamples are indicated in order to classify the relations.

We develop a theory of testing for single phase type processes in Chapter 7 and compare each of the relations defined previously with the notion of testing. Moreover, we prove the strict inclusion of the testing relation in one of the relations of Chapter 6.

Finally, Chapter 8 concludes the thesis and gives directions of further research.

Chapter 2

Background

2.1 Overview

This chapter provides a short introduction to some fundamental mathematical concepts of probability theory which play an important role for the main chapters of this thesis.

We start with a section focusing on Kronecker operations which are used in the sequel.

In Section 2.3, we treat probability spaces and the construction of probability measures. We will resort to these preliminaries in Section 3.6 for a probability measure on sets of paths which is used throughout the subsequent chapters for the probability of certain observations. Basic notations for discrete probability distributions are introduced in the remainder of the first section. Such distributions will emerge in the definition of phase type processes, as randomized scheduler choice, etc. More details can be found in most textbooks on basic probability theory. We refer, for example, to [Fel68].

Section 2.4 shortly illuminates the idea of random variables and their distributions and leads the way for Markov chains (Section 2.5) and phase type distributed random variables (Section 2.6).

The section on Markov chains is essential for the entire thesis: It is explained why Markov chains can be regarded as state-transition graphs. Moreover, the relationship between Markov chains and exponential distributions is highlighted and the parallel composition of Markov chains is covered. The interested reader can consult [Bre99] and [GS82] for more details.

The last section focuses on the definition of phase type distributions and their representations which have been introduced by Neuts [Neu81]. Special cases and closure properties are treated as well as the problem of unique and minimal representations. The section on the theory of phase type distributions points out the many advantages phase type distributions provide. Moreover, it argues for the use of phase type distributions in concurrent stochastic modeling frameworks.

2.2 Kronecker Operations

The following operations are closely related to the parallel composition of CTMCs, closure properties of phase type distributions and parallel composition of PTPs. Let $A(i,j)$ denote an element of the matrix $A \in \mathbb{R}^{n \times m}$ where $1 \leq i \leq n, 1 \leq j \leq m$. The *Kronecker (tensor) product* of two matrices $U \in \mathbb{R}^{n_1 \times m_1}$ and $V \in \mathbb{R}^{n_2 \times m_2}$ is defined as $W = U \otimes V$, $W \in \mathbb{R}^{n_1 n_2 \times m_1 m_2}$ where

$$W((k_1 - 1) \cdot n_2 + k_2, (l_1 - 1) \cdot m_2 + l_2) = U(k_1, l_1) V(k_2, l_2)$$

$(1 \leq k_h \leq n_h, 1 \leq l_h \leq m_h, h \in \{1, 2\})$.

Example 2.1

We consider a simple example with $n_1 = m_1 = 2, n_2 = 3, m_2 = 4$ and

$$U = \begin{pmatrix} u_{11} & u_{12} \\ u_{21} & u_{22} \end{pmatrix} \text{ and } V = \begin{pmatrix} v_{11} & v_{12} & v_{13} & v_{14} \\ v_{21} & v_{22} & v_{23} & v_{24} \\ v_{31} & v_{32} & v_{33} & v_{34} \end{pmatrix}.$$

The tensor product $W = U \otimes V$ is given by

$$W = \begin{pmatrix} u_{11}V & u_{12}V \\ u_{21}V & u_{22}V \end{pmatrix}$$

$$= \left(\begin{array}{cccc|cccc} u_{11}v_{11} & u_{11}v_{12} & u_{11}v_{13} & u_{11}v_{14} & u_{12}v_{11} & u_{12}v_{12} & u_{12}v_{13} & u_{12}v_{14} \\ u_{11}v_{21} & u_{11}v_{22} & u_{11}v_{23} & u_{11}v_{24} & u_{12}v_{21} & u_{12}v_{22} & u_{12}v_{23} & u_{12}v_{24} \\ u_{11}v_{31} & u_{11}v_{32} & u_{11}v_{33} & u_{11}v_{34} & u_{12}v_{31} & u_{12}v_{32} & u_{12}v_{33} & u_{12}v_{34} \\ \hline u_{21}v_{11} & u_{21}v_{12} & u_{21}v_{13} & u_{21}v_{14} & u_{22}v_{11} & u_{22}v_{12} & u_{22}v_{13} & u_{22}v_{14} \\ u_{21}v_{21} & u_{21}v_{22} & u_{21}v_{23} & u_{21}v_{24} & u_{22}v_{21} & u_{22}v_{22} & u_{22}v_{23} & u_{22}v_{24} \\ u_{21}v_{31} & u_{21}v_{32} & u_{21}v_{33} & u_{21}v_{34} & u_{22}v_{31} & u_{22}v_{32} & u_{22}v_{33} & u_{22}v_{34} \end{array}\right).$$

Some important properties of tensor products and sums are

- Associativity: $U \otimes (V \otimes W) = (U \otimes V) \otimes W$
- Distributivity over (ordinary matrix) addition:
  $(U_1 + V_1) \otimes (U_2 + V_2) = (U_1 \otimes U_2) + (V_1 \otimes U_2) + (U_1 \otimes V_2) + (V_1 \otimes V_2)$
- Compatibility with (ordinary matrix) multiplication:
  $(U_1 \times V_1) \otimes (U_2 \times V_2) = (U_1 \otimes U_2) \times (V_1 \otimes V_2)$
- Compatibility with (ordinary matrix) inversion:
  $(U \otimes V)^{-1} = U^{-1} \otimes V^{-1}$

The *Kronecker sum* of two matrices $U \in \mathbb{R}^{n_1 \times n_1}$ and $V \in \mathbb{R}^{n_2 \times n_2}$ is defined by

$$U \oplus V = U \otimes I_{n_2} + I_{n_1} \otimes V$$

where I_n is the identity matrix of size $n \times n$. The index n will be omitted if it is clear from the context.

2.3 Probability Spaces

A *measurable space* is a pair $(\Omega, \mathcal{A})$ such that Ω is a nonempty set of outcomes and $\mathcal{A} \subseteq 2^{\Omega}$ is a sigma-algebra, i.e., $\Omega \in \mathcal{A}$ and $\mathcal{A}$ is closed under countable union and complement. We call Ω the *sample space* and think of $\mathcal{A}$ as the set of all possible events.

For $\mathsf{C} \subseteq 2^{\Omega}$ let $\sigma(\mathsf{C})$ denote the smallest sigma-algebra containing C defined by

$$\sigma(\mathsf{C}) = \bigcap_{\mathcal{A} \supseteq \mathsf{C} \text{ is sigma-algebra on } \Omega} \mathcal{A}.$$

A *probability space* is a tuple $(\Omega, \mathcal{A}, \mathcal{P})$ such that $(\Omega, \mathcal{A})$ is a measurable space and $\mathcal{P} : \mathcal{A} \to [0, 1]$ is a probability measure, i.e., $\mathcal{P}(\Omega) = 1$, $\mathcal{P}(\emptyset) = 0$ and for any countably many pairwise disjoint $A_1, A_2, \ldots \in \mathcal{A}$ we have

$$\sum_i \mathcal{P}(A_i) = \mathcal{P}\big(\bigcup_i A_i\big).$$

Under certain conditions a probability measure defined on a set C can be extended to a unique probability measure on $\sigma(\mathsf{C})$ (for details we refer to [Fel68]).

Example 2.2
Consider $\Omega = [0, 1]$ and $\mathsf{C} = \{(x, y] \mid 0 \leq x < y \leq 1\}$. Then $\sigma(\mathsf{C})$ is the set of all countable unions of closed or open subsets of $[0, 1]$ and $\mathcal{P}((x, y]) = y - x$ can be extended to a unique probability measure (the so-called Lebesgue measure) on $\sigma(\mathsf{C})$.

When dealing with a discrete set Ω, we always consider the power set sigma-algebra 2^Ω and call $\mathcal{P} : 2^\Omega \to [0, 1]$ a *discrete probability measure* (or a *discrete distribution*). For $\omega \in \Omega$ we abbreviate $\mathcal{P}(\{\omega\})$ by $\mathcal{P}(\omega)$. Obviously, $\mathcal{P}$ can be regarded as a function $\mathcal{P} : \Omega \to [0, 1]$ in this case.
Let $\mathsf{dis}(\Omega)$ denote the set of all discrete distributions on Ω. Sometimes we abbreviate $\mathsf{dis}(\Omega)$ by dis_Ω and we say that $\mu \in \mathsf{dis}(\Omega)$ is a *Dirac distribution* if there exists $\omega \in \Omega$ with $\mu(\omega) = 1$ (which implies $\mu(\omega') = 0$ if $\omega' \neq \omega$). We write $\mu = \delta_\omega$ in this case. For a set A we extend $\mu \in \mathsf{dis}(A)$ on set $B \supset A$ by letting $\mu(s) = 0$ if $s \in B \setminus A$
A *discrete subdistribution* over Ω has similar properties as a discrete distribution but does not necessarily sum up to one. We write $\mathsf{sdis}(\Omega)$ for the set of all discrete subdistributions, i.e., functions $\mu : \Omega \to [0, 1]$ with $\sum_{\omega \in \Omega} \mu(\omega) \leq 1$.
The *support* of μ is the set

$$supp(\mu) = \{\omega \in \Omega \mid \mu(\omega) > 0\}$$

and $\mu^\perp$ is defined by

$$\mu^\perp = 1 - \sum_{\omega\in\Omega} \mu(\omega).$$

2.4 Random Variables and Cumulative Distributions

Let $(\Omega, \mathcal{A})$ and $(\Omega', \mathcal{A}')$ be *measurable spaces*. A *random variable* is a function

$$X : (\Omega, \mathcal{A}) \to (\Omega', \mathcal{A}')$$

such that X is $\mathcal{A}$-measurable, i.e., $X^{-1}(A') \in \mathcal{A}$ for all $A' \in \mathcal{A}'$.
If $(\Omega, \mathcal{A}, \mathcal{P})$ is a probability space, then the function P' with

$$\mathcal{P}'(A') = \mathcal{P}(X^{-1}(A')) \quad (\forall A' \in \mathcal{A}') \tag{2.1}$$

defines a probability measure on $(\Omega', \mathcal{A}')$.
If moreover $(\Omega', \mathcal{A}') = (\mathbb{R}, \mathcal{B}(\mathbb{R}))$ where $\mathcal{B}(\mathbb{R})$ is the Borel sigma-algebra (see [Fel68] for details), the *cumulative distribution function* $F_X : \mathbb{R} \to [0, 1]$ is defined as

$$F_X(x) = \mathcal{P}'((-\infty, x]) \quad (\forall x \in \mathbb{R}).$$

In the sequel, we simply refer to F_X as the *distribution of* X. Note that such functions have several useful properties like monotonicity and right-continuity. An instance will be given in Section 2.6 where phase type distributions are introduced.
We may use the term *distribution* for both, discrete distributions and distributions of random variables, if it is clear from the context.

2.5 Continuous Time Markov Chains

Let $\big(X(t)\big)_{t\geq 0}$ be a continuous time stochastic process with a discrete state space, i.e., a family of random variables defined on the same probability space $(\Omega, \mathcal{A}, \mathcal{P})$ and taking values on a discrete set $S = \{s_1, s_2, \ldots\}$. Index t admits the convenient interpretation as time. Thus, $X(t) = s$ means that the process is said to be in state s at time t. We let

$$P(X(t) = s) := P(\{\omega \mid X(t)(\omega) = s\}).$$

According to Equation 2.1 $\big(X(t)\big)_{t\geq 0}$ induces a probability measure $Prob$ with

$$Prob\big(X(t) \in S\big) = Prob\big(\{X(t)(\omega) \in S \mid \omega \in \Omega\}\big) = 1 \quad (\forall t).$$

Process $\big(X(t)\big)_{t\geq 0}$ possesses the *Markov property* if

$$\begin{aligned} &Prob\big(X(t+h) = s_j \mid X(t) = s_i, X(t') = s_{t'}, 0 \leq t' \leq t\big) \\ &= Prob\big(X(t+h) = s_j \mid X(t) = s_i\big) \quad (\forall t, h \geq 0, s_i, s_j \in S). \end{aligned}$$

Informally this means that the future behavior of the process depends only on the current state s_i and not on the history of visited states $s_{t'}$. We call $\big(X(t)\big)_{t\geq 0}$ a *(continuous time) Markov chain* if the Markov property is fulfilled and in case that additionally the *transition probabilities*

$$Prob\big(X(t+h) = s_j \mid X(t) = s_i\big)$$

are independent of t, $\big(X(t)\big)_{t\geq 0}$ is called *homogeneous*, i.e., for a time interval of length h we set

$$p_{ij}(h) := Prob\big(X(t+h) = s_j \mid X(t) = s_i\big) \quad (\forall t).$$

In the following, we will focus on homogeneous continuous time Markov chains, and thus the term Markov chain will always refer to a homogeneous continuous time Markov chain unless otherwise stated.

The time instants at which the process enters a new state are the random variables

$$\begin{aligned} Y(0) &= 0, \\ Y(n+1) &= \inf\{t > Y(n) \mid X(t) \neq X(Y(n))\}, \quad (n \in \{0, 1, \ldots\}) \end{aligned} \tag{2.2}$$

leading to a simple definition of *sojourn times* (or *residence times*)

$$\begin{aligned} D(0) &= 0, \\ D(n) &= Y(n) - Y(n-1), \quad (n \in \{1, 2, \ldots\}). \end{aligned}$$

Thus, the time spent in state $s = X(Y(n))$ is $D(n+1)$.

Obviously, the matrices $P(h) = \big(p_{ij}(h)\big)_{i,j \in \{1,2,\ldots\}}$ and the *initial distribution*

$$\boldsymbol{\nu} := \Big[Prob\big(X(0) = s_1\big), Prob\big(X(0) = s_2\big), \ldots\Big]$$

uniquely determine the *transient state probabilities*

$$\boldsymbol{\nu}(t) := \Big[Prob\big(X(t) = s_1\big), Prob\big(X(t) = s_2\big), \ldots\Big] = \boldsymbol{\nu} \cdot P(t). \tag{2.3}$$

In the sequel, we always use bold letters to refer to (probably infinite) vectors and assume that there is a fixed enumeration of S. For vectors $\boldsymbol{\alpha}, bb, \ldots$ we write $\alpha, \beta, \ldots$ to denote the corresponding discrete distributions.

It is possible to "generate" $P(h)$ from the so-called *generator matrix*

$$Q := \lim_{h \to 0^+} \frac{1}{h}(P(h) - I).$$

It holds that

$$P(h) = \exp(Qh) = \sum_{k=0}^{\infty} \frac{h^k}{k!} Q^k. \tag{2.4}$$

In this thesis, we restrict ourselves to the case that $\sup(-q_{ii}) < \infty$ for all i which can be informally considered as the assumption that a state cannot be left instantaneously. The diagonal entries of the generator matrix are lower or equal zero while the remaining entries are non-negative. Clearly, Equation 2.3 and 2.4 imply that

$$\boldsymbol{\nu}(t) = \boldsymbol{\nu} \exp(Qt) \quad (\forall t).$$

For the calculation of transient state probabilities we refer to [Ste95].

The existence of the generator matrix suggests a characterization of continuous time Markov chains in terms of a state-transition graph with state space S and transitions (s_i, q_{ij}, s_j) for $i \neq j$. The initial distribution is given by incoming edges (without a source) using dashed lines. We call this graphical representation the *intensity graph*. Since $\sum_j q_{ij} = 0$ the diagonal entries can be calculated from the transition labels. Thus, the intensity graph uniquely determines the generator matrix. For the purpose of this thesis, the following definition of Markov chains turns out to be more advantageous.

Definition 2.1 (Continuous Time Markov Chain)
A *(homogeneous) continuous time Markov chain* (CTMC) is a pair $(\boldsymbol{\nu}, Q)$ if there exists an index set $N \subseteq \mathbb{N}$ and a set of states $S = \{s_i \mid i \in N\}$ such that $\nu \in \mathsf{dis}(S)$ and $Q = \big(q_{ij}\big)_{i,j \in N}$ with

i) $q_{ij} \geq 0 \quad \forall i \neq j$,

ii) $\sum_{j \in S} q_{ij} = 0$,

iii) $\sup(-q_{ii}) < \infty$.

Note that if a quadratic matrix Q satisfies condition $i) - iii)$ than it is the generator matrix of a CTMC.
The sojourn times D_n are negative exponentially distributed, i.e.

$$Prob(D_n < h \mid X(Y_{n-1}) = s_i) = 1 - e^{q_{ii}h} \quad (\forall n \in \{1, 2, \ldots\})$$

which justifies the term *exit rate* of state s_i for the entry $-q_{ii}$. Hence, the mean sojourn time in state s_i is $\frac{1}{-q_{ii}}$. The random variables D_n possess the *memoryless property* which is the most striking feature of exponential random variables. It holds that

$$Prob\big(D_n < t + h \mid D_n > t\big) = Prob\big(D_n < h\big) \quad (\forall t \geq 0, h > 0). \tag{2.5}$$

For $-q_{ii} > 0$ and $h > 0$ the time dependent transition probabilities are given by

$$\begin{aligned} & Prob\big(D_n < h \wedge X(Y_n) = s_j \mid X(Y_{n-1}) = s_i\big) \\ = \;& (1 - e^{q_{ii}h})\tfrac{q_{ij}}{-q_{ii}} = \int_0^h q_{ij} e^{q_{ii}t} dt. \end{aligned}$$

Here $\frac{q_{ij}}{-q_{ii}}$ is the probability by which the process enters state s_j after an arbitrary sojourn time in s_i. Thus, $(1 - e^{q_{ii}h})\frac{q_{ij}}{-q_{ii}}$ is the probability of entering state s_j from s_i within h time units.
The non-diagonal entries of Q are referred to as *(transition) rates* because they can be interpreted as the expected number of times the corresponding

transition is taken (per time unit). Assume that random variable Z_k is exponentially distributed with parameter q_{ik} where $k \neq i$ and $q_{ik} > 0$. Then, for a fixed entry $q_{ij} > 0$

$$\begin{aligned} & \int_0^h Prob(Z_j = t, Z_k > t, k \neq j, q_{ik} > 0)dt \\ = & \int_0^h Prob(Z_j = t) \prod_{k:k\neq j, q_{ik}>0} Prob(Z_k > t)dt \\ = & \int_0^h q_{ij} e^{-q_{ij}t} \prod_{k:k\neq j} e^{-q_{ik}t} dt \\ = & \int_0^h q_{ij} e^{q_{ii}t} dt. \end{aligned} \tag{2.6}$$

Intuitively, this means that the time dependent transition probabilities are calculated by assuming that the transition to state s_j (with associated delay Y_j) wins the "race" between the exponential delays Y_k of the possible successor states s_k with $q_{ik} > 0$.

Let $Prob_s$ be the probability measure of Markov chain (δ_s, Q) and let $\big(\hat{X}(t)\big)_{t\geq 0}$ be the corresponding family of random variables. The *(first) recurrence time* of state s is given by random variable $\hat{Z}_s : \Omega \to \mathbb{R}_{>0} \cup \{\infty\}$ with

$$\hat{Z}_s = \inf\{t > \hat{Y}_1 : \hat{X}(t) = s\}$$

where $\hat{Y}$ is the time instant of the first state change as defined in Equation 2.2 but with respect to $\big(\hat{X}(t)\big)_{t\geq 0}$. A state s_i is called *recurrent* if the probability of recurrence in a finite amount of time equals one, i.e., $Prob_s\big(\hat{Z}_s < \infty\big) = 1$. Otherwise s is called *transient*.

Let us now focus on the parallel composition of CTMCs. First recall the definitions of Section 2.2 concerning Kronecker operations. The parallel composition of two finite CTMCs $(\boldsymbol{\nu}_1, Q_1)$ and $(\boldsymbol{\nu}_2, Q_2)$ is defined as $(\boldsymbol{\nu}, Q)$ with

- ⋄ $n = n_1 n_2$ where for $i \in \{1, 2\}$, n_i is the number of states of $(\boldsymbol{\nu}_i, Q_i)$,
- ⋄ $\boldsymbol{\nu} = \boldsymbol{\nu}_1 \otimes \boldsymbol{\nu}_2 \in \mathbb{R}^n$,
- ⋄ $Q = Q_1 \oplus Q_2 \in \mathbb{R}^{n\times n}$.

It is easy to verify that ν is a discrete distribution on $\{1, 2, \dots, n\}$ and Q is a generator matrix. Thus, $(\boldsymbol{\nu}, Q)$ is a CTMC and, if $\boldsymbol{\nu}_i(t)$ contains the transient state probabilities of $(\boldsymbol{\nu}_i, Q_i)$, it can be shown that the parallel composition $(\boldsymbol{\nu}, Q)$ has transient probabilities

$$\boldsymbol{\nu}(t) = \boldsymbol{\nu}_1(t) \otimes \boldsymbol{\nu}_2(t), \quad (\forall t \geq 0).$$

2.6 Phase Type Distributions

Let $\mathbf{1}$ denote the vector with all entries one.

Definition 2.2 (Phase type distribution)
A function $F : \mathbb{R} \to [0, \infty)$ is called a *(continuous) phase type distribution* (PH distribution) if and only if there exists $n \in \{1, 2, \dots\}$, a matrix $T = (T_{ij})_{1 \leq i,j \leq n} \in \mathbb{R}^{n \times n}$ and a row vector $\boldsymbol{\alpha} = [\alpha_1\ \alpha_2\ \dots\ \alpha_n] \in [0, 1]^n$ such that

- ⋄ T is non-singular and for all $i, j \in \{1, \dots, n\}$ with $i \neq j$, $T_{ij} \geq 0$,
- ⋄ $T_{ii} < 0$ and $T_{ii} \leq \sum_{\substack{k=1 \\ k \neq i}}^{n} T_{ik}$ for all $i \in \{1, \dots, n\}$,
- ⋄ $\boldsymbol{\alpha} \cdot \mathbf{1} = \sum_{i=1}^{n} \alpha_i \leq 1$,
- ⋄ $F(x) = \begin{cases} 1 - \boldsymbol{\alpha} \exp(Tx)\mathbf{1} = 1 - \boldsymbol{\alpha}\left(\sum_{k=0}^{\infty} \frac{(Tx)^k}{k!}\right)\mathbf{1} & \text{if } x > 0, \\ 1 - \boldsymbol{\alpha}\mathbf{1} & \text{if } x = 0. \end{cases}$

The pair $(\boldsymbol{\alpha}, T)$ is called a *representation* of F and we say that n is the *order* of $(\boldsymbol{\alpha}, T)$.

Let $F_{(\boldsymbol{\alpha}, T)}$ denote the PH distribution induced by representation $(\boldsymbol{\alpha}, T)$.

A PH distribution is the distribution of the time until absorption in the finite CTMC with generator matrix

$$Q = \begin{bmatrix} T & \mathbf{T^0} \\ \mathbf{0}^\top & 0 \end{bmatrix} \text{ and initial distribution } \boldsymbol{\nu} = \begin{bmatrix} \boldsymbol{\alpha} & (1 - \boldsymbol{\alpha}\mathbf{1}) \end{bmatrix}$$

where $\mathbf{0}^\top = \begin{bmatrix} 0 & \dots & 0 \end{bmatrix}$ is the zero row vector of appropriate size and column vector $\mathbf{T^0} = -T\mathbf{1}$. Matrix T is non-singular if and only if all states T represents are transient [Neu81].

A random variable X with distribution $F_X = F_{(\boldsymbol{\alpha},T)}$ for some $(\boldsymbol{\alpha}, T)$ is called *phase type distributed*. The k-th non-central moment $m_k = E[X^k]$ is then given by

$$m_k = (-1)^k k! \boldsymbol{\alpha} T^{-k} \mathbf{1}. \tag{2.7}$$

Remark 2.1

Let n be the order of $(\boldsymbol{\alpha}, T)$. We can compute the first k moments in an iterative way as follows (compare [Hav98]): In the first step we solve

$$\boldsymbol{\beta}_1 T = -\boldsymbol{\alpha}$$

for $\boldsymbol{\beta}_1 \in \mathbb{R}^n$ which gives $m_1 = \sum_{i=1}^n \beta_1(i)$. Given vector $\boldsymbol{\beta}_1$ we can compute $\boldsymbol{\beta}_2$ and m_2 by

$$\boldsymbol{\beta}_2 T = (-2)\boldsymbol{\beta}_1 \text{ and } m_2 = \sum_{i=1}^n \beta_2(i).$$

In general we have

$$\boldsymbol{\beta}_{k+1} T = -(k+1)\boldsymbol{\beta}_k \text{ and } m_k = \sum_{i=1}^n \beta_k(i).$$

But having T decomposed once in the first step (by using a direct method such as LU-decomposition), the remaining steps can be done in linear time using back-substitutions only. Thus, if direct methods are applicable (because T is small), the first k moments can be computed in time polynomial in the order.

A representation $(\boldsymbol{\alpha}, T)$ of PH distribution is not unique. In general, a PH distribution may have several representations as the following example shows.

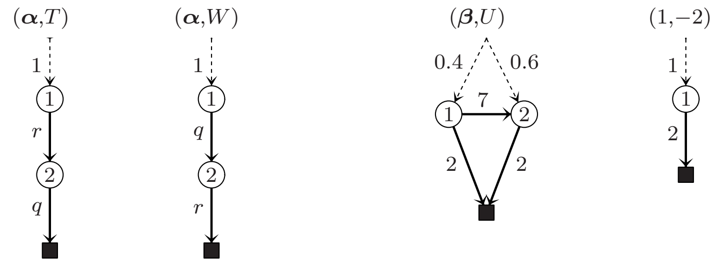

Figure 2.1: Pairs of PH representations which describe the same distribution

Example 2.3
We illustrate phase type representations as follows: Initial distributions are indicated by dashed arrows labeled with probabilities, consecutive phases are connected by solid arrows and edge labels correspond to the phases' rates. The absorbing state is represented by a black square. We consider three instances of phase type representations (see Figure 2.1 for a) and b) and Figure 2.2, left, on page 26 for case c).

a) Let $(\boldsymbol{\alpha}, T)$ and $(\boldsymbol{\alpha}, W)$ be representations of order $n = 2$ with

$$\boldsymbol{\alpha} = \begin{bmatrix} 1 & 0 \end{bmatrix}, T = \begin{bmatrix} -r & r \\ 0 & -q \end{bmatrix} \text{ and } W = \begin{bmatrix} -q & q \\ 0 & -r \end{bmatrix}$$

where $r, q \in \mathbb{R}_{>0}$ and $r \neq q$. It is easy to see that $(\boldsymbol{\alpha}, T)$ and $(\boldsymbol{\alpha}, W)$ represent the same distribution, i.e., $F_{(\boldsymbol{\alpha},T)} = F_{(\boldsymbol{\alpha},W)}$.

b) Let

$$\boldsymbol{\beta} = \begin{bmatrix} 0.4 & 0.6 \end{bmatrix} \text{ and } U = \begin{bmatrix} -9 & 7 \\ 0 & -2 \end{bmatrix}.$$

It can be shown that $F_{(\boldsymbol{\beta},U)} = F_{(1,-2)}$.

c) If $\boldsymbol{\alpha} = 1$ and $T = r < 0$ distribution $F_{(1,r)}$ is the exponential distribution with parameter r.

The following definition can be used to ensure that a representation $(\boldsymbol{\alpha}, T)$ does not contain any superfluous states. More precisely, we remove states that are not reachable from any state i with $\alpha_i > 0$.

Definition 2.3 (Irreducible Representation)
A representation $(\boldsymbol{\alpha}, T)$ is called *irreducible* if and only if each component of the vector $\boldsymbol{\alpha} \exp(Tx)$ is strictly positive for all $x > 0$.

Obviously, removing superfluous states in the underlying absorbing Markov chain such that it becomes irreducible does not change the distribution of the time until absorption.

As Example 2.3 shows, the restriction to irreducible representations does not lead to unique representations of PH distributions, i.e. a PH distribution can have several irreducible representations.

From now on we assume that absorption does not happen immediately, in other words, we consider representations $(\boldsymbol{\alpha}, T)$ with $\boldsymbol{\alpha} \cdot \mathbf{1} = 1$. Furthermore, we assume that representations are irreducible. Let $\mathcal{R}$ be the set of all such representations and $\mathcal{R}_1 \subset \mathcal{R}$ those of order $n = 1$ (which represent exponential distributions). If $(\boldsymbol{\alpha}, T)$ is such that $\boldsymbol{\alpha} = [1\ 0\ \ldots\ 0]$ we simply write T instead of $(\boldsymbol{\alpha}, T)$. For instance, F_{-2} is shorthand for $F_{(1,-2)}$.

Let us now focus on closure properties of phase type distributions.

Definition 2.4 (Finite Mixture)
For $i \in \{1, 2, \ldots, k\}$ let $p_i \in [0, 1]$ be probabilities with $\sum_{i=1}^{k} p_i = 1$. The *finite mixture* of k PH distributions with representations $(\boldsymbol{\alpha}^{(i)}, T^{(i)})$ (of order $n^{(i)}$) is of phase type and has representation $(\boldsymbol{\alpha}, T)$ of order $n = \sum_{i=1}^{k} n^{(i)}$

with

$$\boldsymbol{\alpha} = \begin{bmatrix} p_1\boldsymbol{\alpha}^{(1)} & p_2\boldsymbol{\alpha}^{(2)} & \dots & p_k\boldsymbol{\alpha}^{(k)} \end{bmatrix},\ T = \begin{bmatrix} T^{(1)} & 0 & \cdots & 0 \\ 0 & T^{(2)} & \cdots & 0 \\ \vdots & \vdots & & \vdots \\ 0 & 0 & \cdots & T^{(k)} \end{bmatrix}.$$

Phase type distributions are also closed under *convolution*, for example, the sum of k exponentially distributed random variables is *hypoexponentially distributed* (compare Figure 2.2 on page 26).

Definition 2.5 (Convolution)
The *convolution* of two PH distributions with representations $(\boldsymbol{\beta}, V)$ and $(\boldsymbol{\gamma}, W)$ (of order n and m, respectively) is of phase type and has a representation $(\boldsymbol{\alpha}, T)$ of order $n+m$ with

$$\boldsymbol{\alpha} = \begin{bmatrix} \boldsymbol{\beta} & \mathbf{0}^{\top} \end{bmatrix},\ T = \begin{bmatrix} V & \mathbf{V}^0\boldsymbol{\gamma} \\ 0 & W \end{bmatrix}.$$

We write $F * \hat{F}$ for the convolution of PH distributions F and $\hat{F}$.

It is important to point out that there is a close relationship between the following definition of the minimum of two phase type distributed random variables and the parallel composition of CTMCs. Moreover, the maximum plays an important role in the generalization of the race condition between exponential delays. This is worked out in detail in Chapter 3.

Definition 2.6 (Minimum and Maximum)
The *minimum* and the *maximum* of two phase type distributed random variables with distributions $F(\cdot)$ and $\hat{F}(\cdot)$ are of phase type and have distributions

$$F_{\min}(\cdot) = 1 - \big(1 - F(\cdot)\big)\big(1 - \hat{F}(\cdot)\big) \text{ and } F_{\max}(\cdot) = F(\cdot)\hat{F}(\cdot),$$

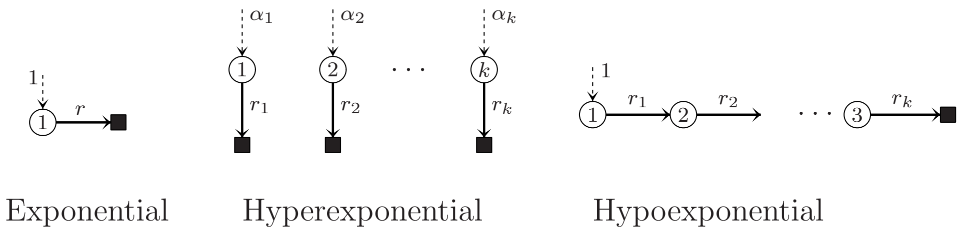

Figure 2.2: Special cases of phase type represenations

respectively. If $(\boldsymbol{\beta}_1, V_1)$ and $(\boldsymbol{\beta}_2, V_2)$ are representations for F and $\hat{F}$ of order n and m, respectively, then $F_{\min}$ has a representation $(\boldsymbol{\alpha}_{\min}, T_{\min})$ of order nm with

$$\boldsymbol{\alpha}_{\min} = \boldsymbol{\beta}_1 \otimes \boldsymbol{\beta}_2,\ T_{\min} = V_1 \oplus V_2$$

and $F_{\max}$ has a representation $(\boldsymbol{\alpha}_{\max}, T_{\max})$ of order $nm + n + m$ with

$$\boldsymbol{\alpha}_{\max} = \begin{bmatrix} \boldsymbol{\beta}_1 \otimes \boldsymbol{\beta}_2 & \mathbf{0}^\top \end{bmatrix},\ T_{\max} = \begin{bmatrix} V_1 \oplus V_2 & I \otimes \mathbf{V}_2^0 & \mathbf{V}_1^0 \otimes I \\ 0 & V_1 & 0 \\ 0 & 0 & V_2 \end{bmatrix}.$$

We now turn to important special cases of phase type distributions: The simplest special case of a phase type distribution is an exponential distribution (compare Example 2.3 c) and Figure 2.2, left). A finite mixture of exponentially distributed random variables is *hyperexponentially distributed*, illustrated in Figure 2.2 (middle) and forms a slightly more flexible subclass. The convolution of exponential distributions results in a hypoexponential distribution (Figure 2.2, right).

Several questions arise when dealing with phase type distributions and their representations.

1. Is there a unique canonical representation of PH distribution F?
2. Can we calculate a minimal representation of F, i.e. a representation $(\boldsymbol{\alpha}, T)$ where T is as small as possible? Here, small means either that

the order of $(\boldsymbol{\alpha}, T)$ is minimal or that T has the smallest number of positive entries.

3. How can we check in an efficient way if two representations correspond to the same distribution?

The first problem has been solved by showing that any PH distribution has a specially structured representation (monocyclic, bi-diagonal, and unicyclic) [CM99, HZ06, HZ05]. The second question is important for the practical use of PH distributions since a smaller representation leads to less computational time. It is still an open problem how to calculate a representation of minimal order. Moreover, even the determination of the minimal order is not possible. For an overview on the theory of PH distributions we refer to O'Cinneide [O'C99], Asmussen [Asm03], Latouche and Ramaswami [LR99b] and the references therein.
The third problem stated above is easy to solve. First note that in general, representations are considerably over-parametrized. A representation of order n has $n^2 + n$ parameters. However, the Laplace transform of the corresponding distribution depends on only $2n$ parameters. The idea is now that, given two representations of order n and m, respectively, we compare the first $2 \cdot \max\{n, m\}$ moments. The following proposition shows that this is sufficient to decide whether they describe the same distribution.

Proposition 2.1

Two representations $(\boldsymbol{\alpha}, T)$ and $(\boldsymbol{\beta}, V)$ of order n and m, respectively, describe the same PH distribution iff their first $2 \cdot \max\{n, m\}$ moments agree.

Proof. We prove the statement by first observing that the distributions of two continuous random variables are equal if and only if their Laplace-Stieltjes transforms are equal. The Laplace-Stieltjes transform of a PH distributed random variable is a fraction $\frac{p(z)}{q(z)}$ of two coprime polynomials. The degree of $p(z)$ is less than or equal to the degree of $q(z)$. Moreover, $q(z)$ has no more than k parameters where k is the order of the smallest representation [O'C99]. Differentiating the Laplace-Stieltjes transform j times and letting $z = 0$ yields the j-th moment. Thus, a PH distribution is uniquely determined by its first $2k$ moments. But then the proposition follows directly. □

Because of Remark 2.1, it takes time polynomial in the order to check whether two representations describe the same distribution.

Chapter 3

Phase Type Processes

3.1 Overview

This section introduces the formalism of phase type processes which are essentially stochastic models for the formal description, specification, and analysis of concurrent systems. They combine the features of two very popular modeling frameworks. On the one hand, the stochastic process underlying a phase type process is a continuous-time Markov chain. On the other hand, a communication structure is provided in the same way as in (action-)labeled transition systems which are the most fundamental models of concurrency. The internal stochastic behavior and the action-based communication structure of phase type processes are clearly separated and yield an operational model which supports both, reasoning about (nondeterministic) communication capabilities and stochastic phenomena.

A phase type process can be represented by a state-transition diagram as this is also possible in the case of Markov chains and labeled transition systems. The versatile class of phase type distributions is used to represent random delays in the system under study. This is realized by transitions labeled by PH representations. If PH representations are absent the process is a probabilistic automaton [Seg95, Sto02]. Another important subclass of phase type processes are *single phase type processes* in which all phase type representations are of order one. Interactive Markov chains [Her02] can be viewed as a special case of a single phase type processes. The definition of phase type processes and useful subclasses are handled in Section 3.2.

The auxiliary construct of a *generator matrix* defined for each pair of states is introduced in Section 3.3. The generator matrix describes the transitions of the Markov chain which underlies the combinations of stochastic state change rules between two states. The definition involves Kronecker operations and uses properties of phase type distributions as given in Chapter 2. It turns out that this matrix representation is very useful for the subsequent chapters. Moreover, it may lead to algorithms similar to those based on matrix-analytic methods for stochastic models (see e.g. [LR99a]). However, analysis algorithms are beyond the scope of this thesis.

It is a widespread approach to give semantics for stochastic models in terms of a probability measure over sets of *paths*. A path of a phase type process captures the behavior of a single realization of the process. We handle paths in Section 3.4.

In Section 3.5 the concept of a *scheduler* is introduced. Schedulers are used to resolve nondeterministic choices which occur in phase type processes. Different types of schedulers are considered which differ in the information about the path history the scheduler's decision depends on and we also distinguish between randomized and deterministic choices. Given a fixed scheduler a probability measure on sets of paths is defined in Section 3.6. All in all, schedulers, paths and probabilities of paths yield a complete picture of the process behavior.

Section 3.7 presents *bisimulation equivalence* for phase type processes by adapting the bisimulation equivalences in [Her02] and [LS91]. As opposed to relations based on the observation of linear sequences (as defined in Chapter 5, 6 and 7) bisimulation respects branching time properties. More precisely, it takes into account that each moment in time may split into various possible futures. In the sequel, bisimulation is used as an intuitive notion of equivalence between PTPs.

Finally, in Section 3.8 we discuss a different approach to instantaneous transitions in PTPs which is based on PH representations that allow a zero delay with positive probability. As opposed to that, in all remaining parts of this thesis we assume that the probability of a phase type distributed delay being non-zero is one.

3.2 Definitions

Let us start with an informal description of the building blocks of a phase type process. Each state of the real-world system under study has a corresponding representative in the phase type process. Since we consider discrete-state models of systems acting in continuous-time, the idea is that state changes occur at discrete points in time (triggered by certain events). We say that the process[1] evolves from one state to another if the current state changes because a transition has been taken. Often, states are identified by a tuple of system variables, e.g. populations of molecular species, numbers of processors being up or down, etc. In the sequel, states are ranged over by s, u, v, w. We propagate primes and indices when necessary.

States of a phase type process have two types of outgoing transitions, namely *phase type transitions* (PH transitions) and *action transitions* both giving the possibility to evolve to a discrete distribution on the set of all states.

PH transitions indicate that a certain amount of time has to pass until the corresponding transition can be taken. This time delay is chosen randomly and follows a phase type distribution where the parameters are given by the transition label. Since the target of a transition is a discrete distribution, the next state is chosen with a certain probability after the transition has been performed. If a state has several outgoing PH transitions for each of them a phase type distributed delay is drawn and the discrete target distribution is chosen according to a *race condition*. More precisely, we generalize the idea of a race between random delays as described in Section 2.5 (compare

[1]The term 'process' is used here for all kinds of state-transition graphs and their compositions.

Equation 2.6 on page 20). The race can be seen as an experiment in which countdown timers are set for each PH transition according to the associated distribution. If the timer of a certain transition expires we say that this transition has become *enabled*. The transition that is enabled before any other "wins" the race and is taken, i.e., the next state is chosen according to the discrete target distribution of that transition and the total amount of time spent in the originating state equals the winner's delay. Since the minimum of phase type distributed random variables is again phase type distributed (compare Definition 2.6 on page 25) the *residence time* in a state (i.e., the time spent in that state) is phase type distributed. We call the phases of the residence time's distribution the *phases of that state*.

Action transitions model process communication and are assumed to happen immediately if communication is possible. They are further distinguished into *visible* and *invisible* ones (the latter are also called *internal* or *hidden [action] transitions*). Visible transitions take place if the process offers an action to its environment. Typically, this feature is used to model process communication. In Section 4.2 we describe how processes can synchronize on action transitions. An invisible transition represents the situation where the process executes some internal operation which is not influenced by the process environment. The target of an action transition is a discrete distribution on the set of all states. Hence, as in case of PH transitions, if an action transition is performed the next state is chosen according to the target distribution of that transition.

The instantaneous execution of a visible action transition might be prevented. Assume that a state has multiple action transitions. Some might be blocked due to certain environment conditions (there is no communication partner available) but others might be already enabled or become enabled after a certain delay because the external environment is already waiting for synchronization on some set of actions or certain actions are externally provided after some time. Invisible transitions are always enabled since they are performed independently from external conditions and thus taken immediately.

This is known as the assumption of *maximal progress* [Her02] and implies that the residence time of a state with at least one outgoing invisible transition is always zero. States with such transitions are called *unstable* or *vanishing* whereas states that do not have an invisible transition are called *stable*.

Besides providing qualitative information about certain events, action transitions impose nondeterminism. More precisely, the process's future splits nondeterministically into multiple alternatives if a state has several enabled action transitions.

We distinguish *internal* and *external nondeterminism*. The former arises from two different ways: From the choice between invisible transitions or from that between visible transitions of the same type, indicated by the same action label.

Internal nondeterminism (also called *pure nondeterminism*) models implementation or scheduling freedom. Its resolution does not depend on external conditions, such as, for example, other processes which are waiting for synchronization. Internal nondeterminism is resolved by the process itself.

External nondeterminism occurs if the process is in a state which has several visible action transitions with different labels. As opposed to internal nondeterminism, external nondeterminism is resolved by the process' reaction on the stimuli of the environment, i.e., the process may proceed in different ways depending on the communication facilities provided by the environment.

Let Act be the set of visible actions ranged over by $a, b, c, \ldots$. Action $\tau \notin \mathsf{Act}$ denotes the distinguished invisible action[2]. By Act_τ we denote the set $\mathsf{Act} \cup \{\tau\}$. Recall that $\mathcal{R}$ is the set of all irreducible phase type representations $(\boldsymbol{\alpha}, T)$ for which $\boldsymbol{\alpha}\mathbf{1} = 1$, i.e., the corresponding delay is greater than zero with probability one. We restrict to this class of representations at this point but discuss the general case in Section 3.8.

---

[2]For the purpose of this thesis it is not necessary to distinguish between several types of invisible actions.

Definition 3.1 (Phase Type Process)
A *phase type process* (PTP) $\mathcal{P}$ is a tuple $(S, \longrightarrow, \twoheadrightarrow, \nu)$ with the following elements:

- S is a non-empty countable *set of states*.
- $\longrightarrow \subseteq S \times \mathcal{R} \times \mathsf{dis}(S)$ is a *phase type transition relation*.
- $\twoheadrightarrow \subseteq S \times \mathsf{Act}_\tau \times \mathsf{dis}(S)$ is an *action transition relation*.
- $\nu \in \mathsf{dis}(S)$ is an *initial distribution*.

For convenience, the generic elements of a PTP $\mathcal{P}$ are denoted by S, $\longrightarrow$, $\twoheadrightarrow$ and ν. We propagate primes and indices when necessary. For example, the elements of a PTP $\mathcal{P}'_i$ are S'_i, $\twoheadrightarrow'_i$, $\longrightarrow'_i$ and ν'_i. If we are dealing with PTPs $\mathcal{P}$ and $\mathcal{Q}$ we may write $\nu_\mathcal{P}$ and $\nu_\mathcal{Q}$ and so on.
We restrict to PTPs that are finitely branching, i.e. the set

$$\big\{(s, (\boldsymbol{\alpha}, T), \mu) \in \longrightarrow \mid (\boldsymbol{\alpha}, T) \in \mathcal{R}, \mu \in \mathsf{dis}(S)\big\} \cup$$
$$\big\{(s, a, \mu) \in \twoheadrightarrow \mid a \in \mathsf{Act}_\tau, \mu \in \mathsf{dis}(S)\big\}$$

is finite for all $s \in S$. This restriction is harmless from the "practical" point of view and simplifies most of the proofs. We claim that many results carry over to processes in which states may have an infinite number of transitions. For simplicity, we also assume the absence of parallel edges, i.e.,

$$s \xrightarrow{a}\!\!\!\!\rightarrow \mu' \wedge s \xrightarrow{a}\!\!\!\!\rightarrow \mu'' \implies \mu' \neq \mu''.$$

We write $s \xrightarrow{a}\!\!\!\!\rightarrow \mu$ if $(s, a, \mu) \in \twoheadrightarrow$ and $s \xrightarrow{\boldsymbol{\alpha}, T} \mu$ if $(s, (\boldsymbol{\alpha}, T), \mu) \in \longrightarrow$. If $\mu(s') = 1$ for some $s' \in S$ we shall also write $s \xrightarrow{a}\!\!\!\!\rightarrow s'$ and $s \xrightarrow{\boldsymbol{\alpha}, T} s'$ for short. Further abbreviations:

- The notation $s \longrightarrow$ means that s has at least one outgoing PH transition. Similarly, we write $s \twoheadrightarrow$ if there is at least one outgoing (visible or invisible) action transition.
- $s \xrightarrow{a}\!\!\!\!\rightarrow$ means that there is at least one outgoing action transition with label $a \in \mathsf{Act}_\tau$.

- We use $s \not\twoheadrightarrow$ to denote that state s has no outgoing action transitions. If s has no outgoing PH transitions we write $s \not\longrightarrow$.
- We say that s is a *deadlock state* if $s \not\longrightarrow$ and $s \not\twoheadrightarrow$.

We assume that if $s \xrightarrow{\tau}\!\!\!\!\twoheadrightarrow$ then $s \not\longrightarrow$. Because of the maximal progress assumption this restriction can be made without loss of generality.

Definition 3.2 (Subclasses)
Let $\mathcal{P} = (S, \longrightarrow, \twoheadrightarrow, \nu)$ be a PTP.

- $\mathcal{P}$ is called a *single phase type process* (SPTP) if all PH representations in $\longrightarrow$ are restricted to those of order $n = 1$, i.e.

$$\longrightarrow \subseteq S \times \mathcal{R}_1 \times \mathsf{dis}(S).$$

  Clearly, this implies that for SPTPs all initial distributions of the PH transitions' labels equal $\boldsymbol{\alpha} = 1$ and the only phase has a single parameter $-r \in \mathbb{R}_{<0}$, i.e., $T = -r$. In this case, we will abbreviate $s \xrightarrow{\boldsymbol{\alpha}, T} \mu$ as $s \xrightarrow{-r} \mu$.
- $\mathcal{P}$ is an *interactive Markov chain* (IMC) if $\mathcal{P}$ is a SPTP and all target distributions of $\twoheadrightarrow$ and $\longrightarrow$ are of Dirac type, i.e. $s \overset{a}{\twoheadrightarrow} \mu$ or $s \xrightarrow{r} \mu$ implies that there exists s' with $\mu = \delta_{s'}$.
- We call $\mathcal{P}$ a *probabilistic automaton* if $\longrightarrow$ is empty.
- $\mathcal{P}$ is a *labeled transition system* (LTS) if it is a probabilistic automaton and all target distributions of $\twoheadrightarrow$ and $\longrightarrow$ are of Dirac type.

In the sequel, we may also mention *continuous-time Markov decision processes* (CTMDPs) which have been studied since the late 1950s [Put94]. A CTMDP can be seen as a special case of an IMC in which the outgoing transitions of a state are either only action transitions or they are exclusively single phased PH transitions. In the former case, internal nondeterminism is absent and deadlock is possible whereas in the latter case at least one

PH transitions must be present. Action and PH states alternate and the initial distribution assigns positive probability to states having only action transitions.

Before we consider an example of a PTP, a short comment on the graphical representation of PTPs is given. PTPs can be represented as graphs. States are depicted by circle nodes and distributions by rectangles. PH transitions and action transitions are drawn using solid edges (in the former case by using a single arrowhead and for the latter case we use two arrowheads) whereas discrete distributions are illustrated by dashed edges. The initial distribution is indicated by incoming edges having no source.

This graphical notation should not be mixed up with the intensity graph of CTMCs. PH transitions with representations of order one are labeled by a single real-valued parameter being *negative*. Although this deviates from the notations used in literature for models such as IMCs, it is appropriate to use $-r$ instead of $r > 0$ in our setting since the parameter is just a special case of a PH representation. Note also that in the case of illustrations of PH representations we depict the intensity graph in which the edges are labeled by the positive entries of the generator matrix. We use thick lines for edges of PH representations and normal lines for PH transitions to point out the difference.

Example 3.1 (Producer/Consumer System)
Figure 3.1 illustrates a PTP which models a producer/consumer system. Here, each state has a pair of labels. The first component represents the current state the producer is in and the second one corresponds to the consumer. Initially, the system starts in state (w_1, w_2) with probability $\nu(w_1, w_2) = 1$ (w is shorthand for "wait"). The visible action a describes the "activation" of the producer, i.e., the producer is activated by obtaining stimulus a from the environment. In state (p_1, w_2) the consumer is still waiting whereas the producer is generating an item (p stands for "production"). This takes X time units where X is PH distributed according to representation $(\boldsymbol{\alpha}, T)$. Distribution μ models the failure probability of production. If the production fails

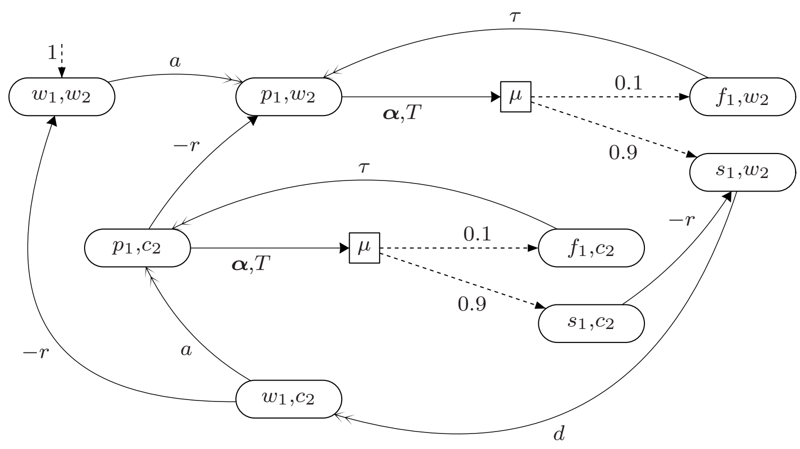

Figure 3.1: A PTP modeling a producer/consumer system.

(f_1) the producer starts again with the production of an item (represented by the τ-transition). With probability 0.9 the production is successful (s_1) and the delivery action d can be performed. After delivery the consumer starts with the consumption of the item which takes an exponentially distributed time (parameter $-r < 0$). However, during consumption (c_2) the producer can be activated once more (indicated by the a-transition emerging from state (w_1, c_2)) and another item can be produced in state (p_1, c_2). Due to the memoryless property of the exponential distribution, state (p_1, c_2) and state (s_1, c_2) have a PH transition which has also label $-r$ (the distribution of the consumption time has again parameter $-r$). Thus, in state (p_1, c_2) there is a race between two PH transitions. Note that the delivery of another item is not possible during consumption in state (s_1, c_2).

3.3 Generator Matrix

The PH transitions of a PTP describe a continuous-time Markov chain if consecutive and competing PH transitions are combined. This property of PTPs is a consequence of the various closure properties of phase type dis-

tributions. Since a Markov chain is completely described by its generator matrix, we can also define the generator matrix of a PTP. It is important to keep in mind that this matrix ignores the action transitions completely. As we will see at the end of this section, the generator matrix will be helpful for the calculation of reachability probabilities.
Recall that for representation $(\boldsymbol{\alpha}, T)$ column vector $\mathbf{T}^0$ contains the absorption rates of the transient states of the underlying absorbing Markov chain.

Definition 3.3 (Generator Matrix)
Let $\mathcal{P}$ be a PTP and let $s \in S$. Assume that s has $k \geq 1$ outgoing PH transitions $s \xrightarrow{\boldsymbol{\alpha}_i, T_i} \mu_i$ where $i \in \{1, 2, \ldots, k\}$ and $(\boldsymbol{\alpha}_i, T_i)$ is of order n_i. Then the number of phases of state s is given by $n_s := \prod_{i=1}^{k} n_i$. Let $\mathbf{1}_{n_i}$ denote the column vector of size n_i with all entries one, and let vector $\mathbf{U}_i$ be defined as

$$\mathbf{U}_i := \Big(\bigotimes_{j=1}^{i-1} \mathbf{1}_{n_j} \Big) \otimes \mathbf{T}_i^0 \otimes \Big(\bigotimes_{j=i+1}^{k} \mathbf{1}_{n_j} \Big) \in \mathbb{R}^{n_s \times 1}.$$

We call row vector

$$\boldsymbol{\gamma}_s := \bigotimes_{i=1}^{k} \boldsymbol{\alpha}_i \in [0,1]^{n_s}$$

the *initial distribution of state* s. The *generator matrix* $Q_{s,s'}$ of the pair (s, s'), $s' \in S$ is a real-valued matrix of size $n_s \times n_{s'}$ with

$$Q_{s,s'} := \begin{cases} \bigoplus_{i=1}^{k} T_i + \sum_{i=1}^{k} \Big(\mathbf{U}_i \cdot \mu_i(s) \cdot \boldsymbol{\gamma}_s \Big) & \text{if } s = s', \\ \sum_{i=1}^{k} \Big(\mathbf{U}_i \cdot \mu_i(s') \cdot \boldsymbol{\gamma}_{s'} \Big) & \text{otherwise.} \end{cases}$$

If $s \not\longrightarrow$ we define $n_s := 1$, $\boldsymbol{\gamma}_s := 1$ and $Q_{s,s'} := \mathbf{0} \in \{0\}^{1 \times n_{s'}}$ for all s'.

The Kronecker sum of the T_i describes the race between the different PH transitions of s, i.e., the entries of $Q_{s,s}$ represent the phases which determine the residence time in s. Assume that $Q_{s,s} \neq 0 \in \mathbb{R}^{1 \times 1}$. Then the residence time distribution in state s is of phase type and given by representation $(\boldsymbol{\gamma}_s, Q_{s,s}) \in \mathcal{R}$. We define $F_s^{res} := F_{(\boldsymbol{\gamma}_s, Q_{s,s})}$ as the residence time distribution of state s if $Q_{s,s} \neq 0$. Otherwise, if $Q_{s,s} = 0$ we let F_s^{res} be a function such that $F_s^{res}(t) = 1$ for all $t \geq 0$. In the sequel, we shortly write Q_s instead of $Q_{s,s}$. If the i-th transition wins the race by passing through the last phase via $\mathbf{U}_i$ then either the process starts again in s (because $\mu_i(s) > 0$) and distributes according to $\boldsymbol{\gamma}_s$ (in this case a PH transition is taken but the target distribution loops back to source s). Or the process evolves from s to $s' \neq s$ according to $\boldsymbol{\gamma}_{s'}$. An instance of a generator matrix is given in Example 3.2.

The initial distribution and the generator matrix of the Markov chain underlying PTP $\mathcal{P}$ is obtained as follows: If ν is the initial distribution of $\mathcal{P}$ and $\boldsymbol{\nu}_s := \nu(s) \cdot \boldsymbol{\gamma}_s$ we define vector $\boldsymbol{\nu}_S := (\boldsymbol{\nu}_s)_{s \in S}$. Similarly, the matrices $Q_{s,s'}$ yield a "global generator matrix": For finite non-empty subsets $A, B \subseteq S$ let $Q_{A,B} := \left(Q_{s,s'}\right)_{s \in A, s' \in B}$ up to enumeration of the elements of A and B. Then $Q_{S,S}$ is the generator matrix of the underlying Markov chain. In the sequel, if A or B are singletons, say $A = \{s\}$, we may write $Q_{s,B}$ instead of $Q_{\{s\},B}$ and we put $Q_{A,s} := Q_{A,\{s\}}$. If $A = B$ we abbreviate $Q_{A,A}$ by Q_A. In a similar way we use the notations $\boldsymbol{\gamma}_A$ and $\boldsymbol{\nu}_A$. We only make use of these notations if the enumeration of the states is not of relevance.

Example 3.2

Consider state s in Figure 3.2 (left) with three outgoing PH transitions and one action transition. The representations that correspond to the PH transitions of s are shown in Figure 3.2 (right). Recall that $\boldsymbol{\gamma}_s$ is the initial distribution of state s (see Definition 3.3). Assume that

$$\boldsymbol{\alpha}_1 = \begin{bmatrix} 0.5 & 0.5 \end{bmatrix}, \quad \boldsymbol{\alpha}_2 = \begin{bmatrix} 0.4 & 0.6 & 0 \end{bmatrix}, \quad \boldsymbol{\alpha}_3 = 1,$$

$$T_1 = \begin{bmatrix} -14 & 10 \\ 0 & -4 \end{bmatrix}, \quad T_2 = \begin{bmatrix} -1 & 1 & 0 \\ 0 & -2 & 2 \\ 0 & 0 & -3 \end{bmatrix}, \quad T_3 = -5,$$

$$\boldsymbol{\gamma}_v = \begin{bmatrix} 0.5 & 0.5 \end{bmatrix}, \quad \boldsymbol{\gamma}_w = \begin{bmatrix} 0.5 & 0.2 & 0.3 \end{bmatrix}.$$

Then the initial distribution of s is given by

$$\boldsymbol{\gamma}_s = \boldsymbol{\alpha}_1 \otimes \boldsymbol{\alpha}_2 \otimes \boldsymbol{\alpha}_3 = \begin{bmatrix} 0.2 & 0.3 & 0 & 0.2 & 0.3 & 0 \end{bmatrix}$$

and for the generator matrix we calculate

$$\begin{aligned} Q_s &= (T_1 \oplus T_2 \oplus T_3) + \mathbf{U}_1 \cdot \mu_1(s) \cdot \boldsymbol{\gamma}_s \\ &= \begin{bmatrix} -19.76 & 1.36 & 0 & 10.24 & 0.36 & 0 \\ 0.24 & -20.64 & 2 & 0.24 & 10.36 & 0 \\ 0.24 & 0.36 & -22 & 0.24 & 0.36 & 10 \\ 0.24 & 0.36 & 0 & -9.76 & 1.36 & 0 \\ 0.24 & 0.36 & 0 & 0.24 & -10.64 & 2 \\ 0.24 & 0.36 & 0 & 0.24 & 0.36 & -12 \end{bmatrix} \end{aligned}$$

$$Q_{s,v} = \begin{bmatrix} 1.4 & 1.4 \\ 1.4 & 1.4 \\ 2.75 & 2.75 \\ 1.4 & 1.4 \\ 1.4 & 1.4 \\ 2.75 & 2.75 \end{bmatrix} \quad Q_{s,w} = \begin{bmatrix} 2.5 & 1 & 1.5 \\ 2.5 & 1 & 1.5 \\ 2.65 & 1.06 & 1.59 \\ 2.5 & 1 & 1.5 \\ 2.5 & 1 & 1.5 \\ 2.65 & 1.06 & 1.59 \end{bmatrix}$$

The race between the three PH transitions of s can be described by a CTMC with $n_s = 6$ transient states, and Q_s is the corresponding part of the generator matrix. Consider, for instance, the third row of Q_s which represents that T_1 and T_3 are in the first and T_2 is in the third phase (state $(1, 3, 1)$ in the parallel composition of the three CTMCs in Figure 3.2, right). T_1 may end up in absorption with rate $\mathbf{T}^0(1) = 4$, enter s again with probability $\mu_1(s) = 0.3$ and start another race in state $(1, 1, 1)$ with probability $\boldsymbol{\gamma}_s(1, 1, 1) = 0.2$. Consequently, the corresponding entry in the first column is $4 \cdot 0.3 \cdot 0.2 = 0.24$. Exit rate -22 shows that state $(1, 3, 1)$ has an absorption rate of

$$22 - (0.24 + 0.36 + 0.24 + 0.36 + 10) = 10.8$$

which is split into $5.5 + 5.3 = 10.8$ because after absorption either v or w is entered. State v is entered with rate

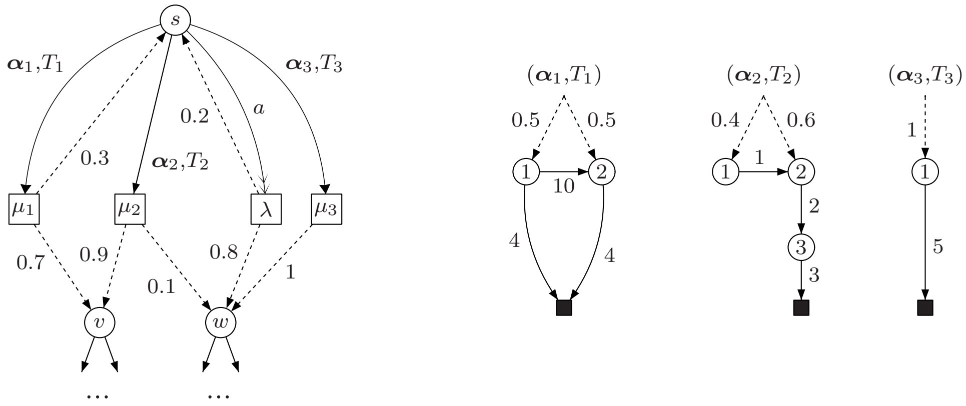

Figure 3.2: State s (left) has one action transition (labeled by a) and three PH transitions with representations $(\boldsymbol{\alpha}_1, T_1)$, $(\boldsymbol{\alpha}_2, T_2)$ and $(\boldsymbol{\alpha}_3, T_3)$ (right).

$$4 \cdot \mu_1(v) + 3 \cdot \mu_2(v) = 4 \cdot 0.7 + 3 \cdot 0.9 = 5.5$$

if T_1 or T_2 wins. In the case of state w we get $3 \cdot 0.1 + 5 \cdot 1 = 5.3$. Matrix $Q_{s,v}$ shows that if state v is entered from $(1, 3, 1)$ with rate 5.5, the initial distribution $\boldsymbol{\gamma}_v$ of v splits this rate into $2.75 + 2.75$ (compare the third row of $Q_{s,v}$).

Let us now concentrate on reachability probabilities of the underlying CTMC of a PTP which can be easily defined by using the generator matrix Q_S. These probabilities will turn out to be very useful, especially for the definition of a probability measure on paths (compare Section 3.6) and for the observation functions in Chapter 5, 6 and 7.

For a PTP $\mathcal{P}$ with state set S, let A, B be finite subsets of S and $J = (x, y] \subseteq \mathbb{R}_{\geq 0}$ with $x < y$. We are interested in the probability to reach some state in B at time instant $t \in J$ by visiting only A-states before t in the Markov chain underlying $\mathcal{P}$. Let us denote this probability by $\mathsf{reach}^{\mathcal{P}}(A, B, J)$. If $B = \emptyset$ or $A = \emptyset$ and $0 \notin J$ we put $\mathsf{reach}^{\mathcal{P}}(A, B, J) = 0$. If $A = \emptyset$ and $0 \in J$

$$\mathsf{reach}^{\mathcal{P}}(A, B, J) := \sum_{s \in B} \nu(s).$$

Otherwise the definition in terms of transient state probabilities is divided into two steps (see also [BHHK03]). The first step calculates the probability of staying within A until time instant x. In the second step we focus on the next $y-x$ time units and the probability to reach B. Recall the definitions of n_s, $\boldsymbol{\nu}_{A'}$ and $Q_{A',B'}$ for $s \in S$, $A', B' \subseteq S$ (compare Definition 3.3 on page 38).

i) Consider the CTMC $(\tilde{\boldsymbol{\nu}}_1, \tilde{Q}_1)$ with $\sum_{s \in A} n_s + 1$ states and

$$\tilde{\boldsymbol{\nu}}_1 := [\boldsymbol{\nu}_A \quad \boldsymbol{\nu}_{S \setminus A}\mathbf{1}],$$

$$\tilde{Q}_1 := \begin{bmatrix} Q_A & Q_{A,S \setminus A}\mathbf{1} \\ \mathbf{0}^\top & 0 \end{bmatrix}.$$

Thus, if set A is left, the process ends up in absorption (the states in $S \setminus A$ are collapsed to one absorbing state). Assume that the transient state probabilities at time instant $x \geq 0$ are given by $\tilde{\boldsymbol{\nu}}_1(x)$. We let

$$[\boldsymbol{\alpha}_{A \setminus B} \quad \boldsymbol{\alpha}_{A \cap B} \quad \alpha_{no}] := \tilde{\boldsymbol{\nu}}_1(x)$$

where we arrange $\tilde{\boldsymbol{\nu}}_1$ according to the partitions $A \setminus B$, $A \cap B$ and $S \setminus A$. We omit an entry if its corresponding set is empty.

ii) Let

$$\tilde{\boldsymbol{\nu}}_2 := [\boldsymbol{\alpha}_{A \setminus B} \quad \boldsymbol{\alpha}_{A \cap B}\mathbf{1} \quad \alpha_{no}],$$

(where we set $\boldsymbol{\alpha}_{A \cap B}\mathbf{1} = 0$ if $A \cap B = \emptyset$ but $B \setminus A \neq \emptyset$). Furthermore, we define $C := A \setminus B$ and

$$\tilde{Q}_2 := \begin{bmatrix} Q_C & Q_{C,B}\mathbf{1} & Q_{C,S \setminus (A \cup B)}\mathbf{1} \\ \mathbf{0}^\top & 0 & 0 \\ \mathbf{0}^\top & 0 & 0 \end{bmatrix}.$$

Then $(\tilde{\boldsymbol{\nu}}_2, \tilde{Q}_2)$ is a CTMC with $2 + \sum_{s \in C} n_s$ states in which all states of B and $S \setminus (A \cup B)$ are absorbing. Note that if set A is not left within $[0, x]$, the process proceeds either in $A \setminus B$ with $\boldsymbol{\alpha}_{A \setminus B}$ or has already reached $A \cap B$ with probability $\boldsymbol{\alpha}_{A\cap B}\mathbf{1}$. If A is left before x (which happens with probability α_{no}) the process remains in the final state. Let

$$\tilde{\boldsymbol{\nu}}_2(y-x) =: [\boldsymbol{\beta}_C \quad \beta_{yes} \quad \beta_{no}]$$

be the transient state probability vector at time instant $y - x$.

Probability $\mathsf{reach}^{\mathcal{P}}(A, B, J)$ is then given by

$$\mathsf{reach}^{\mathcal{P}}(A, B, J) = \beta_{yes}. \tag{3.1}$$

Let $s \in S$ for a fixed PTP $\mathcal{P}$. We may write $\mathsf{reach}^{\mathcal{P}}_s(A, B, J)$ for the probability to reach some state in B at time instant $t \in J$ by visiting only A-states before t while starting in s at time instant 0 with probability one. Obviously, if $\nu(s) = 1$ then $\mathsf{reach}^{\mathcal{P}}_s(A, B, J) = \mathsf{reach}^{\mathcal{P}}(A, B, J)$. We may omit superscript $\mathcal{P}$ if it is clear from the context.

Example 3.3

Assume that state s of Example 3.2 has initial probability 0.8. Thus,

$$\boldsymbol{\nu}_s = 0.8 \cdot \boldsymbol{\gamma}_s = \begin{bmatrix} 0.16 & 0.24 & 0 & 0.16 & 0.24 & 0 \end{bmatrix}.$$

We are interested in the probability $\mathsf{reach}_s(A, B, J)$ with $A = \{s\}$, $B = \{v\}$, $J = (0.2, 0.3]$. We get

$$\tilde{\boldsymbol{\nu}}_1 = \begin{bmatrix} 0.16 & 0.24 & 0 & 0.16 & 0.24 & 0 & 0.2 \end{bmatrix}$$

and

$$\tilde{Q}_1 = \begin{bmatrix} Q_s & Q_{s,\{v,w\}}\mathbf{1} \\ \mathbf{0}^\top & 0 \end{bmatrix}.$$

The transient state probabilities at time instant 0.2 are given by

$$\begin{aligned} \tilde{\boldsymbol{\nu}}_1(0.2) &= \tilde{\boldsymbol{\nu}}_1 \cdot \exp(\tilde{Q}_1 \cdot 0.2) \\ &\approx \begin{bmatrix} 0.006 & 0.008 & 0.002 & 0.050 & 0.071 & 0.022 & 0.840 \end{bmatrix}. \end{aligned}$$

This implies that $\alpha_{no} \approx 0.84$. Since $A \cap B = \emptyset$, $C = \{s\}$ and $S \setminus (A \cup B) = \{w\}$ the initial distribution of the second step is given by

$$\tilde{\boldsymbol{\nu}}_2 := \begin{bmatrix} 0.006 & 0.008 & 0.002 & 0.050 & 0.071 & 0.022 & 0 & 0.840 \end{bmatrix}$$

and generator matrix $\tilde{Q}_2$ is given by

$$\tilde{Q}_2 = \begin{bmatrix} Q_s & Q_{s,v}\mathbf{1} & Q_{s,w}\mathbf{1} \\ \mathbf{0}^\top & 0 & 0 \\ \mathbf{0}^\top & 0 & 0 \end{bmatrix}.$$

Finally, we calculate transient state probabilities at time instant 0.1

$$\begin{aligned} \tilde{\boldsymbol{\nu}}_2(0.1) &= \tilde{\boldsymbol{\nu}}_2 \cdot \exp\left(\tilde{Q}_2 \cdot 0.1\right) \\ &\approx \begin{bmatrix} 0.002 & 0.003 & 0.001 & 0.022 & 0.030 & 0.012 & 0.035 & 0.895 \end{bmatrix} \end{aligned}$$

and derive $\mathsf{reach}_s(A, B, J) \approx 0.035$.

3.4 Paths

Informally, an *(execution) path* of a PTP corresponds to a single realization of the process, i.e. , we can think of a simulation experiment which captures the process' behavior in time. Initially, a random number is drawn to decide for an initial state according to the initial distribution. Afterwards, the process evolves from state to state according to the transition relations. If the current state, say s, is stable (which means that $s \not\xrightarrow{\tau}\!\!\!\twoheadrightarrow$) the process resides in s as long as none of the outgoing transitions is enabled. If s has at least one outgoing transition, the first one which is enabled is taken. The successor state is determined by the transition's target distribution, say μ, i.e., another random variable is used to decide for a state in the support of μ. In case that s is unstable, one of its outgoing τ transitions is selected to reach the next state. This alternation of state and transition selection proceeds infinitely long if there is always at least one enabled transition and ends in a state otherwise.

We use simplifying environment conditions at this point which will be relaxed in subsequent chapters. Actions are either immediately available or assumed to be completely blocked, i.e., if s has an outgoing transition labeled by a visible action there are three possible continuations of a path: Either the action transition is taken immediately, or the paths ends in s (because no other transition becomes enabled), or a PH transition is taken (if there is any).

Here, each state change is accompanied by an *event*. An event is either

1. the entering of the next state via a PH transition after remaining in the current state for a non-zero duration, or
2. the instantaneous execution of an action.

The first type of event is represented by the residence time whereas in the second case only the performed action is recorded. Let $\mathsf{E} = \mathbb{R}_{>0} \cup \mathsf{Act}_\tau$ be the set of events.

Definition 3.4 (Paths and path fragments)
Let $s_1, s_2, \ldots, s_k, \ldots$ be states of a PTP $\mathcal{P}$ and let $\mathsf{e}_1, \mathsf{e}_2, \ldots, \mathsf{e}_{k-1}, \ldots \in \mathsf{E}$. A *path* of $\mathcal{P}$ is an infinite or finite sequence

$$\pi = s_1 \xrightarrow{\mathsf{e}_1} s_2 \xrightarrow{\mathsf{e}_2} \ldots \quad \text{or} \quad \pi = s_1 \xrightarrow{\mathsf{e}_1} s_2 \xrightarrow{\mathsf{e}_2} \ldots \xrightarrow{\mathsf{e}_{k-1}} s_k$$

such that for $i \in \{1, 2, \ldots\}$ ($i < k$ if π is finite) there exists $\mu_i \in \mathsf{dis}(S)$ with $\mu_i(s_{i+1}) > 0$ and

1. $\mathsf{e}_i \in \mathbb{R}_{>0}$ implies $s_i \xrightarrow{\boldsymbol{\alpha}, T} \mu_i$ for some $(\boldsymbol{\alpha}, T)$ and s_i is stable,
2. $\mathsf{e}_i = a \in \mathsf{Act}_\tau$ implies $s_i \overset{a}{\twoheadrightarrow} \mu_i$.

We require maximality in the following sense: If π is finite then $s_k \not\longrightarrow$ and s_k is stable. The intuition behind this is as follows: If $s_k \longrightarrow$ the process will leave state s_k after a finite amount of time with probability one. If s_k is unstable, it is left immediately. Therefore, the only case in which the process may remain in s_k forever is the case where s_k is stable and has no outgoing PH transition.

A *path fragment* is a prefix of a path that ends in a state. Let $\mathsf{path}(s)$ and $\mathsf{pathf}(s)$ denote the set of paths and path fragments, respectively, that start in state s. If PTP $\mathcal{P}$ has initial distribution ν we let

$$\mathsf{path}(\mathcal{P}) = \bigcup_{\nu(s)>0} \mathsf{path}(s) \text{ and } \mathsf{pathf}(\mathcal{P}) = \bigcup_{\nu(s)>0} \mathsf{pathf}(s)$$

In the sequel, we will use the following notations for paths and path fragments:

- By $\pi{\downarrow}_i$ we denote path fragment $\xi = s_1 \xrightarrow{\mathsf{e}_1} s_2 \xrightarrow{\mathsf{e}_2} \ldots \xrightarrow{\mathsf{e}_{i-1}} s_i$ of path π. If π is finite $\pi{\downarrow}_i$ is only defined for $i \leq k$. We may sometimes abbreviate ξ by $s_1\mathsf{e}_1 s_2\mathsf{e}_2 \ldots \mathsf{e}_{i-1}s_i$.
- Let $last(\xi)$ and $first(\xi)$ denote the last state and the first state on path fragment ξ, respectively, i.e. if

$$\xi = s_1 \xrightarrow{\mathsf{e}_1} s_2 \xrightarrow{\mathsf{e}_2} \ldots \xrightarrow{\mathsf{e}_{i-1}} s_i$$

  we have $last(\xi) = s_i$ and $first(\xi) = s_1$.
- $time(\xi)$ denotes the total amount of time that has passed on path fragment ξ, i.e.,

$$time(\xi) = \sum_{1\leq j<i} t_j \text{ where } t_j = \begin{cases} \mathsf{e}_j & \text{if } \mathsf{e}_j \in \mathbb{R}_{>0}, \\ 0 & \text{otherwise.} \end{cases}$$

- Let $trace(\xi) \in \mathsf{Act}^*$ denote the ordered sequence of visible actions on ξ and we put

$$\mathsf{pathf}(\mathcal{P}, \sigma) := \{\xi \in \mathsf{pathf}(\mathcal{P}) \mid trace(\xi) = \sigma\}.$$

  By ϵ we denote the empty sequence, i.e. if ξ contains no visible action we write $trace(\xi) = \epsilon$.
- Let $|\xi| = |\pi{\downarrow}_i| = i - 1 \in \{0, 1, \ldots\}$ denote the number of events of ξ.

- The time-abstract copy $untime(\xi) \in (S \cup \mathsf{Act}_\tau)^*$ of path fragment ξ is obtained by omitting all residence times of ξ. For consecutive elements of $untime(\xi)$ we write $s_i \longrightarrow s_{i+1}$ if $\mathsf{e}_i \in \mathbb{R}_{>0}$ and if $\mathsf{e}_i = a$ we keep writing $s_i \xrightarrow{a} s_{i+1}$. Clearly, $trace(\xi)$ depends only on $\kappa = untime(\xi)$. Therefore, we may also write $trace(\kappa)$ instead of $trace(\xi)$. Similarly, $last(\kappa) := last(\xi)$, $first(\kappa) := first(\xi)$.

Example 3.4
An example of a finite path (fragment) of the PTP which models the producer/consumer system (see Example 3.1 on page 36) is

$$\pi = (w_1, w_2) \xrightarrow{a} (p_1, w_2) \xrightarrow{1.1} (f_1, w_2) \xrightarrow{\tau} (p_1, w_2) \xrightarrow{2.5} (s_1, w_2).$$

Then $\pi\downarrow_3 = (w_1, w_2) \xrightarrow{a} (p_1, w_2) \xrightarrow{1.1} (f_1, w_2)$, $last(\pi) = (s_1, w_2)$ and $time(\pi) = 1.1 + 2.5 = 3.6$. Furthermore, $untime(\pi)$ is the sequence

$$(w_1, w_2) \xrightarrow{a} (p_1, w_2) \longrightarrow (f_1, w_2) \xrightarrow{\tau} (p_1, w_2) \longrightarrow (s_1, w_2)$$

and $trace(\pi) = a$.

Not all paths of a PTP are of interest: an infinite number of transitions taken in a finite amount of time contradicts our intuition of a reactive system operating in continuous time. This situation is referred to as *zeno behavior* and occurs, for instance, if the PTP contains a cycle consisting of action transitions. For the rest of this thesis, we restrict to PTPs that do not show zeno behavior.

A further restriction is made concerning *divergence*. In the sequel, we require that all considered PTPs are *divergence free* meaning that there exists no infinite path which contains an infinite number of consecutive τ-transitions.

The above restrictions are made to simplify most of the proofs. We claim that all results are still valid if we replace these restrictions by an adaption of Segala's *probabilistic convergence*. However, the proofs would become more technical.

3.5 Schedulers

This section focuses to the concept of *schedulers* [Var85], also often called *policies* [Put94, dA97] or *adversaries* [Seg95]. They are used to resolve nondeterministic decisions. In general, internal or external nondeterminism is present in a PTP. A scheduler gives priorities to the outgoing transitions of a state and the behavior of the process relative to the scheduler becomes fully probabilistic. Thus, for a fixed scheduler $\mathcal{D}$, each path occurs with a certain probability. More precisely, a probability space can be defined where the sample space is the set of all paths that are possible under $\mathcal{D}$. Until now, we have not yet given a formal semantics for PTPs but discussed only informally their meaning. In the sequel, we define the their formal semantics based on the decisions of a scheduler.

The detailed functioning of a scheduler $\mathcal{D}$ can be informally described as follows: Let s be a state of PTP $\mathcal{P}$. Scheduler $\mathcal{D}$ can either decide that

- ⋄ state s is left immediately by selecting an action transition among the ones available in state s,
- ⋄ or, if s is stable, $\mathcal{D}$ decides that s is not left immediately (because none of the action transitions is already enabled) and a race between the outgoing PH transitions of s starts.

If $s \not\longrightarrow$ and s is stable, scheduler $\mathcal{D}$ has also the possibility to choose a non-zero deadlock probability, i.e., the process remains in s for an infinite amount of time.

Definition 3.5 (Scheduler)
A *scheduler* for PTP $\mathcal{P}$ is a function

$$\mathcal{D} : \mathsf{pathf}(\mathcal{P}) \rightarrow \mathsf{sdis}(\mathsf{Act}_\tau \times \mathsf{dis}_S)$$

such that for $\xi \in \mathsf{pathf}(\mathcal{P})$ with $last(\xi) = s$ and $untime(\xi) = \kappa$

i) $s \xrightarrow{\tau}\!\!\!\!\rightarrow$ implies $\mathcal{D}(\xi) \in \mathsf{dis}(\mathsf{Act}_\tau \times \mathsf{dis}_S)$,

ii) $\mathcal{D}(\xi)(a,\mu) > 0$ implies $s \xrightarrow{a}\!\!\!\!\rightarrow \mu$.

iii) There exists a partition of $\mathbb{R}_{\geq 0}$ into countably many pairwise disjoint, non-empty intervals $I_1, I_2, \ldots \subseteq \mathbb{R}_{\geq 0}$ such that for all $\xi_1, \xi_2 \in \mathsf{pathf}(\mathcal{P})$ with $untime(\xi_1) = untime(\xi_2) = \kappa$ it holds that:
If $time(\xi_1), time(\xi_2) \in I_i$ for some $i \in \{1, 2, \ldots\}$ then $\mathcal{D}(\xi_1) = \mathcal{D}(\xi_2)$.

Let us examine the three conditions given above. The first condition states that if $s \xrightarrow{\tau}\!\!\!\!\rightarrow$ the residence time in s is zero and with probability one, s is left immediately. In the case of condition ii) we require that the pair (a, μ) has a non-zero probability only if $s \xrightarrow{a}\!\!\!\!\rightarrow \mu$. Condition iii) ensures that $\mathcal{D}$ behaves nicely in the sense that $\mathcal{D}$'s decisions are piecewise constant with respect to the time elapsed in previous steps. This avoids measurability problems.
In our setting, schedulers do not resolve the (probabilistic) choice between PH transitions. If $\lambda = \mathcal{D}(\xi)(a,\mu) \in \mathsf{sdis}(\mathsf{Act}_\tau \times \mathsf{dis}_S)$ then, with probability

$$\lambda^\perp = 1 - \sum_{(a,\mu): s \xrightarrow{a}\!\!\!\!\rightarrow \mu} \lambda(a, \mu),$$

no action transition is selected at all. If additionally $s \longrightarrow$, the race between the outgoing PH transitions takes place with probability $\lambda^\perp$. Note that if $s \not\!\!\longrightarrow\!\!\!\!\rightarrow$ the race between the outgoing PH transitions of s takes place with probability $\lambda^\perp = 1$.

Remark 3.1
Consider an extension of the above definition in which schedulers give priorities to certain action transitions but may also cause that action transitions are enabled after a certain delay. More precisely, in case that the scheduler decides for a non-zero residence time in s, a race between PH distributed delays starts. For a (possibly empty) subset of the set of action transitions of s, PH distributions are chosen. Each of these representations defines the distribution of the amount of time that has to elapse until the corresponding action transition is enabled. Thus, there is a race between the PH transitions

and (some of) the action transitions of s. All remaining action transitions are assumed to be blocked completely. We are not going to analyze such schedulers at this point but refer to chapter 6 for this class of schedulers.

Let us consider several subclasses of schedulers because often the full class of schedulers is too powerful. One might be interested in the behavior of PTP $\mathcal{P}$ with respect to a certain type of scheduler. We follow the standard classification of schedulers similar to that in [Put94] for Markov decision processes.

Definition 3.6 (Scheduler Types)
Let $\mathcal{D}$ be a scheduler for PTP $\mathcal{P}$. $\mathcal{D}$ is called

- *time-abstract* if $\mathcal{D}$'s decisions do not depend on the time elapsed in previous steps, i.e., for all path fragments ξ, ξ', if $untime(\xi) = untime(\xi')$ then $\mathcal{D}(\xi) = \mathcal{D}(\xi')$,
- *deterministic* if $\mathcal{D}$ only schedules transitions with probability zero or one, i.e., $\mathcal{D}$ can be regarded as a partial function

$$\mathsf{pathf}(\mathcal{P}) \to (\mathsf{Act}_\tau \times \mathsf{dis}_S),$$

- *stationary* if $\mathcal{D}$'s choice depends only on the current state, i.e., $\mathcal{D}(\xi) = \mathcal{D}(\xi')$ for all $\xi, \xi' \in \mathsf{pathf}(\mathcal{P})$ with $last(\xi) = last(\xi')$,
- *total* if for each ξ with $last(\xi) \not\twoheadrightarrow$ and $last(\xi) \longrightarrow$ the choice $\mathcal{D}(\xi)$ is a distribution on S which prevents $\mathcal{D}$ from deciding for deadlock although $last(\xi)$ has outgoing action transitions.

We use the following abbreviations. Let THR be the set of all schedulers. THD denotes the subclass of deterministic schedulers, while HR stands for the set of all time-abstract schedulers and HD for its subclass of time-abstract deterministic schedulers. We write SR for the subclass of HR consisting of all stationary schedulers and SD $\subset$ SR$\cap$HD for the set of all stationary, deterministic schedulers. Thus, T stands for "time dependent", R for "randomized", D for "deterministic", S for "stationary" and H for "history-dependent". The prefix t will we used to denote that we consider total schedulers. E.g., tHD means the class of all total HD-schedulers. Moreover, for

$$D \in \{\mathsf{THR}, \mathsf{THD}, \mathsf{HR}, \mathsf{HD}, \mathsf{SR}, \mathsf{SD}, \mathsf{tTHR}, \ldots, \mathsf{tSD}\}$$

we may write $D(\mathcal{P})$ to denote the set of all D-scheduler for PTP $\mathcal{P}$. Consequently, $\mathsf{THR}(\mathcal{P})$ denotes the set of all schedulers for $\mathcal{P}$.
In the case of HR-schedulers we may write $\mathcal{D}(\kappa)$ instead of $\mathcal{D}(\xi)$ if $\kappa = \mathit{untime}(\xi)$. If $\mathcal{D}$ is a SR-scheduler $\mathcal{D}(\xi)$ is abbreviated by $\mathcal{D}(s)$ where $s = \mathit{last}(\xi)$. Similarly, for a choice $\delta_{(a,\mu)}$ of a deterministic scheduler $\mathcal{D}$ we write $\mathcal{D}(\xi) = (a, \mu)$.

3.6 Path Probabilities

In this section, we present the semantics of a PTP in terms of paths and probabilities of paths.
Let $\mathcal{D}$ be a scheduler for PTP $\mathcal{P}$, i.e. $\mathcal{D} \in \mathsf{THR}(\mathcal{P})$. A *$\mathcal{D}$-path* is a (finite or infinite) path $\pi \in \mathsf{path}(\mathcal{P})$ with

$$\pi = s_1 \xrightarrow{\mathsf{e}_1} s_2 \xrightarrow{\mathsf{e}_2} \ldots \text{ or } \pi = s_1 \xrightarrow{\mathsf{e}_1} s_2 \xrightarrow{\mathsf{e}_2} \ldots \xrightarrow{\mathsf{e}_{k-1}} s_k$$

that obeys $\mathcal{D}$'s decisions, i.e., for all $i \geq 1$ ($i < k$ if π is finite) there exists $\mu_i \in \mathsf{dis}(S)$ with $\mu_i(s_{i+1}) > 0$ and whenever

$$\mathsf{e}_i = \begin{cases} t \in \mathbb{R}_{>0} & \text{then } \mathcal{D}(\pi{\downarrow_i}) = \lambda \text{ with } \lambda^{\perp} > 0, \\ a \in \mathsf{Act}_{\tau} & \text{then } \mathcal{D}(\pi{\downarrow_i}) = \lambda \text{ with } \lambda(a, \mu_i) > 0. \end{cases}$$

A *$\mathcal{D}$-path fragment* is a path fragment of a $\mathcal{D}$-path.

Now, recall the definition and construction of probability spaces in Section 2.3 on page 14. Scheduler $\mathcal{D}$ induces a probability space as follows: The sample space Ω is the set of all $\mathcal{D}$-paths, a sigma-algebra and a probability measure is constructed using the standard cylinder set construction for CTMDP-like models (see e.g. [Put94]).

Let

$$\zeta = s_1 \ E_1 \ s_2 \ E_2 \dots \ E_{k-1} \ s_k$$

where $s_1, s_2, \dots, s_k \in S$ and for $i \in \{1, 2, \dots, k-1\}$ either $E_i = \{a\}$ for some $a \in \mathsf{Act}_\tau$ or $E_i = (x, y] \subseteq \mathbb{R}_{>0}$, $x < y$, and $\mathcal{D}$'s decisions are constant on $(x, y]$ (recall condition iii) of Definition 3.5 on page 48). Let $\Xi_\zeta^\mathcal{D}$ be the set of $\mathcal{D}$-path fragments ξ such that

$$\xi = s_1 \xrightarrow{\mathsf{e}_1} s_2 \xrightarrow{\mathsf{e}_2} \dots \xrightarrow{\mathsf{e}_{k-1}} s_k$$

and $\mathsf{e}_i \in E_i$ for all $i \in \{1, 2, \dots, k-1\}$. Then *cylinder set* $\mathsf{C}_\zeta^\mathcal{D}$ is the set of all $\mathcal{D}$-paths π with $\pi{\downarrow}_k \in \Xi_\zeta^\mathcal{D}$. Hence, $\mathsf{C}_s^\mathcal{D}$ is the set of all $\mathcal{D}$-paths that start in s. Let $\mathcal{C}^\mathcal{D}$ be the set of all such cylinder sets $\mathsf{C}_\zeta^\mathcal{D}$. Set $\mathcal{C}^\mathcal{D}$ yields a basis for the σ-algebra $\Sigma^\mathcal{D} := \sigma(\mathcal{C}^\mathcal{D})$ and a probability measure $\mathsf{Pr}^\mathcal{D}$ on $\Sigma^\mathcal{D}$ is defined by specifying the probabilities for the elements of $\mathcal{C}^\mathcal{D}$.

For $k = 1$ we define

$$\mathsf{Pr}^\mathcal{D}(\mathsf{C}_s^\mathcal{D}) := \nu(s)$$

where ν is the initial distribution of $\mathcal{P}$. Now, let $k > 1$, $\pi \in \mathsf{C}_\zeta^\mathcal{D}$ with

$$\begin{aligned} \pi{\downarrow}_k &= s_1 \xrightarrow{\mathsf{e}_1} s_2 \xrightarrow{\mathsf{e}_2} \dots \xrightarrow{\mathsf{e}_{k-1}} s_k, \\ \zeta &= s_1 \ E_1 \ s_2 \ E_2 \dots E_{k-1} \ s_k, \\ \zeta' &:= s_1 \ E_1 \ s_2 \ E_2 \dots E_{k-2} \ s_{k-1}, \end{aligned}$$

and $\mathcal{D}(\pi{\downarrow}_{k-1}) =: \lambda$. Then λ depends only on ζ' since $\mathcal{D}$ makes the same decision for all path fragments in $\Xi_{\zeta'}$. We distinguish two cases:

1. Assume that $E_{k-1} \subseteq \mathbb{R}_{>0}$. This implies that $\lambda^\perp > 0$ and a race between the outgoing PH transitions of s_{k-1} determines the probability to evolve from state s_{k-1} to s_k within $t \in E_{k-1}$ time units. We set

$$\mathsf{Pr}^\mathcal{D}(\mathsf{C}_\zeta^\mathcal{D}) = \mathsf{Pr}^\mathcal{D}(\mathsf{C}_{\zeta'}^\mathcal{D}) \cdot \lambda^\perp \cdot \mathsf{reach}_{s_{k-1}}(\{s_{k-1}\}, \{s_k\}, E_{k-1}).$$

2. Now, assume that $E_{k-1} = \{a\}$, $a \in \mathsf{Act}_\tau$. In this case, s_k is entered immediately and we put

$$\mathsf{Pr}^\mathcal{D}(\mathsf{C}_\zeta^\mathcal{D}) = \mathsf{Pr}^\mathcal{D}(\mathsf{C}_{\zeta'}^\mathcal{D}) \cdot \sum \{\!|\lambda(a, \mu) \cdot \mu(s_k) \mid \exists \mu : s_{k-1} \xrightarrow{a} \mu|\!\}$$

where $\{\!|\dots|\!\}$ denotes a multi-set.

In the subsequent chapters, the unique extension of $\mathsf{Pr}^{\mathcal{D}}$ on the complete sigma-algebra $\Sigma^{\mathcal{D}}$ is denoted by $\mathsf{Pr}^{\mathcal{D}}$ as well. Sometimes we may add subscript $\mathcal{P}$, i.e. we write $\mathsf{Pr}^{\mathcal{D}}_{\mathcal{P}}$ instead of $\mathsf{Pr}^{\mathcal{D}}$. Moreover, we may abbreviate

$$\mathsf{Pr}^{\mathcal{D}}(\{\pi \in \mathsf{path}(\mathcal{P}) \mid \exists i : \pi{\downarrow_i}= \xi\})$$

by $\mathsf{Pr}^{\mathcal{D}}(\xi)$. We may take as sample space the set of all paths instead of the set of $\mathcal{D}$-paths by assuming that sets of paths that are prohibited by $\mathcal{D}$ have probability zero.
Note that the above probability measure also enables us to reason about the minimal and maximal probability of certain sets of paths by ranging over all schedulers or certain sets of schedulers.

3.7 Phase Type Bisimulation

The idea of bisimulation is that the behavior of a PTP can be mimicked stepwise by an equivalent one. Here, a "step" corresponds to the execution of a single action transition or to that of a PH transition.
For the remainder of this section, assume for simplicity that there is no state that is in the support of a discrete distribution being the target of a PH transition emerging from that state (PH self-loops). All results presented in the sequel are also valid without this restriction but the proofs would be less readable.

Let us first fix some notations. For PTP $\mathcal{P}$ we write $s \xrightarrow{a}\!\!\!\!\rightarrow{}^{c} \mu$ iff there exist $p_1, p_2, \dots, p_n \in [0, 1]$ such that $\sum_{i=1}^{n} p_i = 1$, $\mu = \sum_{i=1}^{n} p_i\mu_i$ and $s \xrightarrow{a}\!\!\!\!\rightarrow \mu_i$ for each $i \in \{1, 2, \dots, n\}$. We call $s \xrightarrow{a}\!\!\!\!\rightarrow{}^{c} \mu$ a *combined transition*.
For a given equivalence relation R on a set S and $\mu_1, \mu_2 \in \mathsf{dis}(S)$, we write $\mu_1 \equiv_R \mu_2$ iff for all equivalence classes $C \in S/R$

$$\sum_{s\in C} \mu_1(s) = \sum_{s\in C} \mu_2(s).$$

Definition 3.7 (Phase Type Bisimulation)
Let $\mathcal{P}_1$ and $\mathcal{P}_2$ be PTPs with $S_1 \cap S_2 = \emptyset$. An equivalence relation R on $S_1 \cup S_2$ is a *phase type bisimulation* between $\mathcal{P}_1$ and $\mathcal{P}_2$ iff $(s_1, s_2) \in R$ implies that

i) for all $a \in \mathsf{Act}_\tau$ whenever $s_1 \xrightarrow{a} \mu_1$ in either $\mathcal{P}_1$ or $\mathcal{P}_2$ then there exists a combined transition $s_2 \xrightarrow{a}{}^c \mu_2$ in either $\mathcal{P}_2$ or $\mathcal{P}_1$ such that $\mu_1 \equiv_R \mu_2$,

ii) for all $C \in S/R$, $C \neq [s_1]_R$, $t \geq 0$

$$reach_{s_1}(\{s_1\}, C, [0,t]) = reach_{s_2}(\{s_2\}, C, [0,t]),$$

iii) s_1 and s_2 have the same residence time distribution, i.e.

$$F^{res}_{s_1} = F^{res}_{s_2}.$$

We write $\mathcal{P}_1 =_{\mathsf{bs}} \mathcal{P}_2$ iff there exists a phase type bisimulation R between $\mathcal{P}_1$ and $\mathcal{P}_2$ such that for the initial distributions holds $\nu_1 \equiv_R \nu_2$.
Note that $reach_s(\{s\}, C, [0,t])$ is the probability to reach a state in C within t time units via a single PH transition from s. For a phase type bisimulation $R \subseteq S \times S$ let $s_1 R s_2$ and

$$\{[s_1]_R, C_1, C_2, \ldots\} = S/R$$

where S/R denotes the quotient space and $[s_1]_R$ the equivalence class of s_1. For $i \in \{1,2\}$ let $n_i := n_{s_i}$ be the number of phases of s_i, $A_i := [s_1]_R \setminus \{s_i\}$ and[3]

$$Q_i := \begin{bmatrix} Q_{s_i} & Q_{s_i,A_i}\mathbf{1} & Q_{s_i,C_1}\mathbf{1} & Q_{s_i,C_2}\mathbf{1} & \ldots \\ \mathbf{0}^\top_{n_i} & 0 & 0 & 0 & \ldots \\ \mathbf{0}^\top_{n_i} & 0 & 0 & 0 & \ldots \\ \vdots & \vdots & \vdots & \vdots & \end{bmatrix},$$

$$V_i := \begin{bmatrix} \mathbf{1}_{n_i} & \mathbf{0}_{n_i} & \mathbf{0}_{n_i} & \ldots \\ 0 & 1 & 0 & \ldots \\ 0 & 0 & 1 & \ldots \\ \vdots & \vdots & \ddots & \ddots \end{bmatrix}$$

[3]Note that the case $A_i = \emptyset$ is not relevant here, because it implies that $s_1 = s_2$.

where $\mathbf{0}$ ($\mathbf{1}$) is the column vector with zeros (ones, respectively) of appropriate size (of size $n \in \mathbb{N}$ if subscript n is added) and superscript $\top$ refers to the transposed vector. Then Condition ii) and iii) are equivalent to

$$\boldsymbol{\gamma}_{s_1} e^{Q_1 t} V_1 = \boldsymbol{\gamma}_{s_2} e^{Q_2 t} V_2, \quad \forall t \geq 0$$

because vector $\boldsymbol{\gamma}_{s_i} e^{Q_i t} V_i$ contains the probability to enter C_j (A_i, respectively) before time instant t for all $j \in \{1, 2, \ldots\}$. The first entry is the probability to stay in state s_i until time instant t. Note that Condition ii) and iii) specify a multivariate PH distribution [ALS84] for which $(\boldsymbol{\gamma}_{s_i}, Q_i)$ is a representation (if we omit zero-entries and use the fact that $\mathcal{P}_1$ and $\mathcal{P}_2$ are finitely branching). Thus, for all $C \in S/R$, $J = (x, y] \subset R_{\geq 0}$, $x < y$, the probability to reach C within time interval J from s_i via a single transition is

$$reach_{s_1}(\{s_1\}, C, J) = reach_{s_2}(\{s_2\}, C, J). \tag{3.2}$$

In case of PTPs in which all target distributions of PH transitions are of type Dirac, we have that ii) and iii) hold iff $F(s_1, C) = F(s_2, C)$ for all $C \in S/R$ where $F(s, C)$ is the distribution of the time to reach an element of C from s (via a single transition). It is easy to see that $F(s_i, C), C \neq [s_1]_R$ is a PH distribution if s_i has at least one PH transition leading to an element of C. Formally,

$$F(s_i, C) = \begin{cases} \perp & \text{if } \nexists u \in C : s_i \xrightarrow{\boldsymbol{\alpha}, T} u, \\ 1 - \prod\limits_{s_i \xrightarrow{\boldsymbol{\alpha}, T} u, u \in C} \big(1 - F_{(\boldsymbol{\alpha}, T)}\big) & \text{otherwise.} \end{cases}$$

Note that according to Definition 2.6 on page 25, a representation of $F(s_i, C) \neq \perp$ is given by

$$\boldsymbol{\beta} = \bigotimes_{s_i \xrightarrow{\boldsymbol{\alpha}, T} u, u \in C} \boldsymbol{\alpha}, \qquad U = \bigoplus_{s_i \xrightarrow{\boldsymbol{\alpha}, T} u, u \in C} T.$$

In the case of IMCs this reduces to the sum of the rates of all transitions leading to C. Therefore, the above definition is a natural extension of the

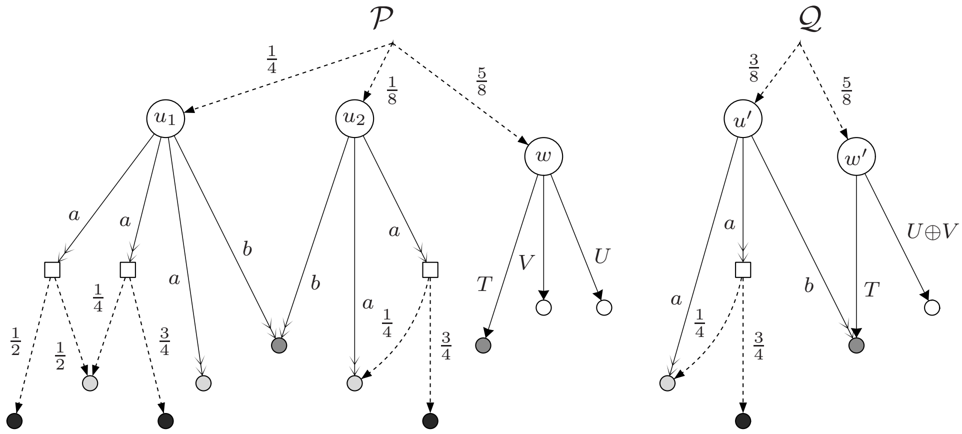

Figure 3.3: $\mathcal{P}$ and $\mathcal{Q}$ are bisimulation equivalent.

strong bisimulation in [Her02]. More precisely, two IMCs $\mathcal{P}$, $\mathcal{Q}$ are in relation $=_{\mathsf{bs}}$ iff they are strongly bisimilar according to [Her02][4].

Example 3.5

Consider two PTPs $\mathcal{P}$ and $\mathcal{Q}$ (partly illustrated in Figure 3.3) and assume that equivalence relation R identifies those states that are depicted on the same height such as, for example, u_1, u_2 and u' or w and w'. Additionally, the successors of u_1, u_2, w, u' and w' are shaded according to the equivalence classes they belong to (their outgoing transitions are omitted in the figure). It is easy to see that all combined a-transitions of u_1 can be matched by a combined a-transition of u' (or u_2) and vice versa. The probability to reach the dark shaded equivalence class via an a-transition is always between 0 and $\frac{3}{4}$ and that of reaching the light shaded class is between $\frac{1}{4}$ and 1. The same holds for the b-transitions. We have that w and w' are related because with the same probability the white (the gray) equivalence class is reached within t time units and the residence time in w and w' is PH distributed with representation $T \oplus U \oplus V$.

[4]Recall that in contrast to [Her02] we do not allow PH self-loops. However, it is possible to modify the definition of phase type bisimulation such that PH self-loops are taken into account.

Let us treat the relationship between phase type bisimulation and the semantics of PTPs in terms of path probabilities. For simplicity, we restrict to time-abstract schedulers and treat the time-dependent case afterwards.

Let $\mathcal{P}$ and $\mathcal{Q}$ be PTPs with $\mathcal{P} =_{\mathsf{bs}} \mathcal{Q}$ and let R be a phase type bisimulation which relates $\mathcal{P}$ and $\mathcal{Q}$. For $k \geq 1$ let

$$\eta = A_1 \; E_1 \; A_2 \; E_2 \ldots E_{k-1} \; A_k$$

where for $j \in \{1, 2, \ldots, k-1\}$ either $E_j = (x, y] \subseteq \mathbb{R}_{>0}$, $x < y$, or $E_j = \{a\}$ for some $a \in \mathsf{Act}_\tau$ and $A_1, A_2, \ldots, A_k \subseteq S_\mathcal{P} \cup S_\mathcal{Q}$ are equivalence classes of R. We define Ξ_η as the set of all path fragments $\xi \in \mathsf{pathf}(\mathcal{P}) \cup \mathsf{pathf}(\mathcal{Q})$ such that $|\xi| = k$, the l-th state on ξ is in A_l for all $l \in \{1, 2, \ldots, k\}$, the j-th event is in E_j for all j. The length $|\eta|$ is defined as the number of equivalence classes, i.e. $|\eta| = k$. By $\mathcal{H}^R$ we denote the set of all such sets Ξ_η.

Lemma 3.1

Let $\mathcal{P}, \mathcal{Q}$ be PTPs with $\mathcal{P} =_{\mathsf{bs}} \mathcal{Q}$ and let R be a phase type bisimulation which relates $\mathcal{P}$ and $\mathcal{Q}$. Then for all $\mathcal{D} \in \mathsf{HR}(\mathcal{P})$ there exists $\mathcal{D}' \in \mathsf{HR}(\mathcal{Q})$ such that for all $k \geq 1$, $\Xi_\eta \in \mathcal{H}^R$, $|\eta| = k$

$$\mathsf{Pr}_\mathcal{P}^\mathcal{D}(\{\pi \in \mathsf{path}(\mathcal{P}) \mid \pi{\downarrow_k} \in \Xi_\eta\}) = \mathsf{Pr}_\mathcal{Q}^{\mathcal{D}'}(\{\pi \in \mathsf{path}(\mathcal{Q}) \mid \pi{\downarrow_k} \in \Xi_\eta\}).$$

Proof. Let $\eta = A_1 \; E_1 \; A_2 \; E_2 \ldots \; E_{k-1} \; A_k$. We first decompose

$$\mathsf{Pr}_\mathcal{P}^\mathcal{D}(\Xi_\eta) \;\; := \;\; \mathsf{Pr}_\mathcal{P}^\mathcal{D}(\{\pi \in \mathsf{path}(\mathcal{P}) \mid \pi{\downarrow_k} \in \Xi_\eta\})$$

into summands

$$\mathsf{Pr}_\mathcal{P}^\mathcal{D}(\Xi_\zeta) \;\; := \;\; \mathsf{Pr}_\mathcal{P}^\mathcal{D}(\{\pi \in \mathsf{path}(\mathcal{P}) \mid \pi{\downarrow_k} \in \Xi_\zeta\})$$

where $\zeta = s_1 \; E_1 \; s_2 \; E_2 \ldots \; E_{k-1} \; s_k$ and $s_i \in A_i$ for $1 \leq i \leq k$. Let $Z(\eta, \mathcal{P})$ be the set of all such ζ for which additionally $s_i \in S_\mathcal{P}$ for all i. We use the same notation for $\mathcal{Q}$. Since $\mathcal{D}$ is time-abstract we may also write $\mathcal{D}(\zeta)$ instead of $\mathcal{D}(\xi)$ whenever $\xi \in \Xi_\zeta$ (and $last(\zeta)$ instead of $last(\xi)$). Furthermore, we observe that for each $\zeta \in Z(\eta, \mathcal{P})$ and each $a \in \mathsf{Act}_\tau$ scheduler $\mathcal{D}$ induces a combined transition $s \xrightarrow{a}\!\!\!\!\rightarrow^c \mu$ with $s = last(\zeta)$ and

$$\mu(u) = \sum_{\mu' : s \xrightarrow{a}\!\!\!\!\rightarrow \mu'} \frac{\mathcal{D}(\zeta)(a, \mu')}{\mathcal{D}(\zeta, a)} \cdot \mu'(u) \quad (u \in S_{\mathcal{P}})$$

provided $\mathcal{D}(\zeta, a) := \sum_{\mu' : s \xrightarrow{a}\!\!\!\!\rightarrow \mu'} \mathcal{D}(\zeta)(a, \mu') > 0$. We write $\mathcal{D}(\zeta) \xrightarrow{a}\!\!\!\!\rightarrow \mu$ in this case. However, if $\mathcal{D}(\zeta, a) = 0$ no such combined transition exists. Vice versa, if $\mathcal{D}(\zeta) \xrightarrow{a}\!\!\!\!\rightarrow \mu$, the choices $\mathcal{D}(\zeta)(a, \mu')$ are uniquely determined by μ and $\mathcal{D}(\zeta, a)$. Additionally, we define

$$\begin{aligned} \mathsf{Pr}^{\mathcal{D}}_{\mathcal{P}}(\Xi_\zeta, a) &:= \mathsf{Pr}^{\mathcal{D}}_{\mathcal{P}}(\Xi_\zeta) \cdot \mathcal{D}(\zeta, a) \\ \mathsf{Pr}^{\mathcal{D}}_{\mathcal{P}}(\Xi_\eta, a) &:= \textstyle\sum_{\zeta \in Z(\eta, \mathcal{P})} \mathsf{Pr}^{\mathcal{D}}_{\mathcal{P}}(\Xi_\zeta, a). \end{aligned}$$

and use the same notations for $\mathcal{D}'$ and $\zeta \in Z(\eta, \mathcal{Q})$.

We construct $\mathcal{D}' \in \mathsf{HR}(\mathcal{Q})$ as follows: Let $\xi' \in \mathsf{pathf}(\mathcal{Q})$, let $\Xi_\eta \in \mathcal{H}^R$ be such that $\xi' \in \Xi_\eta$. We define $\mathcal{D}'(\xi')(a, \mu') = 0$ for all $\mu' \in \mathsf{dis}(\mathcal{Q})$ if $\mathcal{D}(\zeta, a) = 0$ for all $\zeta \in Z(\eta, \mathcal{P})$. If $\mathcal{D}(\zeta, a) > 0$ for some $\zeta \in Z(\eta, \mathcal{P})$ then $\mathcal{D}(\zeta) \xrightarrow{a}\!\!\!\!\rightarrow \mu_\zeta$. We know from $last(\xi')\, R\, last(\zeta)$ that for each such ζ there exists a combined transition $last(\xi') \xrightarrow{a}\!\!\!\!\rightarrow^c \mu'_\zeta$ with $\mu'_\zeta \equiv_R \mu_\zeta$. Assume that $\mathsf{Pr}^{\mathcal{D}}_{\mathcal{P}}(\Xi_\eta) > 0$ (otherwise $\mathcal{D}'(\xi')$ is defined arbitrary). We choose $\mathcal{D}'$ such that

$$\mathcal{D}'(\xi') \xrightarrow{a}\!\!\!\!\rightarrow \sum_{\zeta \in Z(\eta, \mathcal{P})} \frac{\mathsf{Pr}^{\mathcal{D}}_{\mathcal{P}}(\Xi_\zeta, a)}{\mathsf{Pr}^{\mathcal{D}}_{\mathcal{P}}(\Xi_\eta, a)} \cdot \mu'_\zeta$$

and

$$\mathcal{D}'(\eta, a) := \textstyle\sum_{\zeta \in Z(\eta, \mathcal{P})} \frac{\mathsf{Pr}^{\mathcal{D}}_{\mathcal{P}}(\Xi_\zeta, a)}{\mathsf{Pr}^{\mathcal{D}}_{\mathcal{P}}(\Xi_\eta)} = \sum_{\zeta \in Z(\eta, \mathcal{P})} \frac{\mathsf{Pr}^{\mathcal{D}}_{\mathcal{P}}(\Xi_\zeta)}{\mathsf{Pr}^{\mathcal{D}}_{\mathcal{P}}(\Xi_\eta)} \cdot \mathcal{D}(\zeta, a)$$

Thus, the choice of $\mathcal{D}'$ for a is determined by a convex combination of the combined transitions induced by $\mathcal{D}$ and a convex combination of the probability that $\mathcal{D}$ decides for a. It is important to note that $\mathcal{D}'(\eta, a)$ depends only on η and *not* on ξ' whereas $\mathcal{D}'(\xi')(a,\mu)$ depends on ζ' where $\xi' \in \Xi_{\zeta'}$ (we may write $\mathcal{D}'(\zeta')$ instead of $\mathcal{D}'(\xi')$).

We now proceed by induction on $|\eta|$ and show that $\mathsf{Pr}^{\mathcal{D}}_{\mathcal{P}}(\Xi_\eta) = \mathsf{Pr}^{\mathcal{D}'}_{\mathcal{Q}}(\Xi_\eta)$. Assume that $|\eta| = 1$, i.e. there exists an equivalence class A with $\Xi_\eta = \Xi_A$. But then by using $\nu_{\mathcal{P}} \equiv_R \nu_{\mathcal{Q}}$ we get

$$\begin{aligned}\mathsf{Pr}^{\mathcal{D}}_{\mathcal{P}}(\Xi_A) &= \textstyle\sum_{s\in A} \mathsf{Pr}^{\mathcal{D}}_{\mathcal{P}}(s) = \sum_{s\in A} \nu_{\mathcal{P}}(s)\\ &= \textstyle\sum_{s\in A} \nu_{\mathcal{Q}}(s) = \sum_{s\in A} \mathsf{Pr}^{\mathcal{D}'}_{\mathcal{Q}}(s) = \mathsf{Pr}^{\mathcal{D}'}_{\mathcal{Q}}(\Xi_A).\end{aligned}$$

We now turn to the induction step. Let $\eta' = A_1\ E_1\ A_2\ E_2 \dots E_k\ A_{k+1}$ and $\eta = A_1\ E_1\ A_2\ E_2 \dots\ E_{k-1}\ A_k$. We distinguish two cases: $E_k = \{a\}, a \in \mathsf{Act}_\tau$ and $E_k \subseteq \mathbb{R}_{>0}$. Assume that $E_k = \{a\}$. It holds that for all $\zeta' \in Z(\eta, \mathcal{Q})$

$$\begin{aligned}\mathcal{D}'(\eta, a, A_{k+1}) &:= \sum_{\mu: last(\zeta') \xrightarrow{a} \mu} \mathcal{D}'(\zeta')(a,\mu)\cdot \mu(A_{k+1})\\ &= \mathcal{D}'(\eta,a)\cdot \textstyle\sum_{\zeta\in Z(\eta,\mathcal{P})} \frac{\mathsf{Pr}^{\mathcal{D}}_{\mathcal{P}}(\Xi_\zeta,a)}{\mathsf{Pr}^{\mathcal{D}}_{\mathcal{P}}(\Xi_\eta,a)}\cdot \mu'_\zeta(A_{k+1})\\ &= \mathcal{D}'(\eta,a)\cdot \textstyle\sum_{\zeta\in Z(\eta,\mathcal{P})} \frac{\mathsf{Pr}^{\mathcal{D}}_{\mathcal{P}}(\Xi_\zeta,a)}{\mathsf{Pr}^{\mathcal{D}}_{\mathcal{P}}(\Xi_\eta,a)}\cdot \mu_\zeta(A_{k+1}),\end{aligned}$$

i.e. $\mathcal{D}'(\eta, a, A_{k+1})$ is independent of ζ'. Therefore we get[5]

$$\begin{aligned}\mathsf{Pr}^{\mathcal{D}'}_{\mathcal{Q}}(\Xi_{\eta'}) &= \sum_{\zeta'\in Z(\eta,\mathcal{Q})} \mathsf{Pr}^{\mathcal{D}'}(\Xi_{\zeta'})\cdot \sum_{\mu: last(\zeta') \xrightarrow{a} \mu} \mathcal{D}'(\zeta')(a,\mu)\cdot\mu(A_{k+1})\\ &= \mathsf{Pr}^{\mathcal{D}'}(\Xi_\eta)\cdot \mathcal{D}'(\eta,a,A_{k+1})\\ &= \mathsf{Pr}^{\mathcal{D}'}(\Xi_\eta)\cdot \mathcal{D}'(\eta,a)\cdot \sum_{\zeta\in Z(\eta,\mathcal{P})} \frac{\mathsf{Pr}^{\mathcal{D}}(\Xi_\zeta,a)}{\mathsf{Pr}^{\mathcal{D}}(\Xi_\eta,a)}\cdot\mu_\zeta(A_{k+1})\\ &= \mathsf{Pr}^{\mathcal{D}'}(\Xi_\eta)\cdot \sum_{\hat\zeta\in Z(\eta,\mathcal{P})} \frac{\mathsf{Pr}^{\mathcal{D}}(\Xi_{\hat\zeta},a)}{\mathsf{Pr}^{\mathcal{D}}(\Xi_\eta)}\cdot \sum_{\zeta\in Z(\eta,\mathcal{P})} \frac{\mathsf{Pr}^{\mathcal{D}}(\Xi_\zeta,a)}{\mathsf{Pr}^{\mathcal{D}}(\Xi_\eta,a)}\cdot\mu_\zeta(A_{k+1})\\ &\overset{\text{ind. hyp.}}{=} \sum_{\hat\zeta\in Z(\eta,\mathcal{P})} \frac{\mathsf{Pr}^{\mathcal{D}}(\Xi_{\hat\zeta},a)}{\mathsf{Pr}^{\mathcal{D}}(\Xi_\eta,a)}\cdot \sum_{\zeta\in Z(\eta,\mathcal{P})} \mathsf{Pr}^{\mathcal{D}}(\Xi_\zeta,a)\cdot\mu_\zeta(A_{k+1}).\\ &= \sum_{\zeta\in Z(\eta,\mathcal{P})} \mathsf{Pr}^{\mathcal{D}}(\Xi_\zeta,a)\cdot\mu_\zeta(A_{k+1}).\\ &= \mathsf{Pr}^{\mathcal{D}}_{\mathcal{P}}(\Xi_{\eta'}).\end{aligned}$$

[5] We sometimes omit subscript $\mathcal{P}$ and $\mathcal{Q}$ and write $\mathsf{Pr}^{\mathcal{D}}$ and $\mathsf{Pr}^{\mathcal{D}'}$ to improve readability.

Finally, assume that E_k is an interval. Let $\zeta' \in Z(\eta, \mathcal{Q})$ and $last(\zeta') = s'$. We define

$$\mathcal{D}'(\zeta', E_k, A_{k+1}) := \mathcal{D}'(\zeta')^{\perp} \cdot \mathsf{reach}_{s'}(\{s'\}, A_{k+1}, E_k).$$

From condition ii) of Definition 3.7 and Equation 3.2 (see page 55) we know that sRs' implies $\mathsf{reach}_s(\{s\}, A_{k+1}, E_k) = \mathsf{reach}_{s'}(\{s'\}, A_{k+1}, E_k)$. Combining this with the fact that

$$\mathcal{D}'(\zeta')^{\perp} = \sum_{\zeta \in Z(\eta,\mathcal{P})} \frac{\mathsf{Pr}^{\mathcal{D}}(\Xi_\zeta)}{\mathsf{Pr}^{\mathcal{D}}(\Xi_\eta)} \cdot \mathcal{D}(\zeta)^{\perp} \quad (\forall \zeta' \in Z(\eta, \mathcal{Q}))$$

yields for all $\zeta' \in Z(\eta, \mathcal{Q})$

$$\begin{aligned}
\mathcal{D}'(\zeta', E_k, A_{k+1}) &= \mathcal{D}'(\zeta')^{\perp} \cdot \mathsf{reach}_{last(\zeta')}(\{last(\zeta')\}, A_{k+1}, E_k) \\
&= \sum_{\zeta \in Z(\eta,\mathcal{P})} \frac{\mathsf{Pr}^{\mathcal{D}}(\Xi_\zeta)}{\mathsf{Pr}^{\mathcal{D}}(\Xi_\eta)} \cdot \mathcal{D}(\zeta)^{\perp} \cdot \mathsf{reach}_{last(\zeta')}(\{last(\zeta')\}, A_{k+1}, E_k) \\
&= \sum_{\zeta \in Z(\eta,\mathcal{P})} \frac{\mathsf{Pr}^{\mathcal{D}}(\Xi_\zeta)}{\mathsf{Pr}^{\mathcal{D}}(\Xi_\eta)} \cdot \mathcal{D}(\zeta)^{\perp} \cdot \mathsf{reach}_{last(\zeta)}(\{last(\zeta)\}, A_{k+1}, E_k) \\
&=: \sum_{\zeta \in Z(\eta,\mathcal{P})} \frac{\mathsf{Pr}^{\mathcal{D}}(\Xi_\zeta)}{\mathsf{Pr}^{\mathcal{D}}(\Xi_\eta)} \cdot \mathcal{D}(\zeta, E_k, A_{k+1})
\end{aligned}$$

But then

$$\begin{aligned}
\mathsf{Pr}^{\mathcal{D}'}_{\mathcal{Q}}(\Xi_{\eta'}) &= \sum_{\zeta' \in Z(\eta,\mathcal{Q})} \mathsf{Pr}^{\mathcal{D}'}(\Xi_{\zeta'}) \cdot \mathcal{D}'(\zeta', E_k, A_{k+1}) \\
&= \mathsf{Pr}^{\mathcal{D}'}(\Xi_\eta) \cdot \sum_{\zeta \in Z(\eta,\mathcal{P})} \frac{\mathsf{Pr}^{\mathcal{D}}(\Xi_\zeta)}{\mathsf{Pr}^{\mathcal{D}}(\Xi_\eta)} \cdot \mathcal{D}(\zeta, E_k, A_{k+1}) \\
&\overset{\text{ind. hyp.}}{=} \sum_{\zeta \in Z(\eta,\mathcal{P})} \mathsf{Pr}^{\mathcal{D}}(\Xi_\zeta) \cdot \mathcal{D}(\zeta, E_k, A_{k+1}) \\
&= \mathsf{Pr}^{\mathcal{D}}_{\mathcal{P}}(\Xi_{\eta'})
\end{aligned}$$

which concludes the case $E_k \subseteq \mathbb{R}_{>0}$ and the lemma follows. □

The above result can be extended to the case of time-dependent schedulers:

Theorem 3.1

Let $\mathcal{P}, \mathcal{Q}$ be PTP*s with $\mathcal{P} =_{\mathsf{bs}} \mathcal{Q}$ and let R be a phase type bisimulation which relates $\mathcal{P}$ and $\mathcal{Q}$. For all $\mathcal{D} \in \mathsf{THR}(\mathcal{P})$ there exists $\mathcal{D}' \in \mathsf{THR}(\mathcal{Q})$ such that for all $k \geq 1$, $\Xi_\eta \in \mathcal{H}^R$, $|\eta| = k$*

$$\mathsf{Pr}_{\mathcal{P}}^{\mathcal{D}}(\{\pi \in \mathsf{path}(\mathcal{P}) \mid \pi{\downarrow_k} \in \Xi_\eta\}) = \mathsf{Pr}_{\mathcal{Q}}^{\mathcal{D}'}(\{\pi \in \mathsf{path}(\mathcal{Q}) \mid \pi{\downarrow_k} \in \Xi_\eta\}).$$

Proof. Assume that $\mathcal{D} \in \mathsf{THR}(\mathcal{P})$. The construction of a scheduler $\mathcal{D}' \in \mathsf{THR}(\mathcal{Q})$ such that for all $\Xi_\eta \in \mathcal{H}^R$

$$\mathsf{Pr}_{\mathcal{Q}}^{\mathcal{D}'}(\Xi_\eta) = \mathsf{Pr}_{\mathcal{P}}^{\mathcal{D}}(\Xi_\eta)$$

is exactly the same as for the time-independent case (see Lemma 3.1) except that we first prove the statement for the case that $\mathcal{D}$'s choice is constant on all intervals of η. For all remaining sets Ξ_η we proceed as follows: Let $I_1, I_2, \ldots \subseteq \mathbb{R}_{\geq 0}$ be the intervals on which $\mathcal{D}$'s choice is constant (recall condition iii of Definition 3.5). We define $\mathcal{D}'$ such that its choice is constant on exactly the same partition of $\mathbb{R}_{\geq 0}$ and use the fact that we can write Ξ_η as the disjoint union of sets $\Xi_{\eta'}$ where $\mathcal{D}$'s choice is constant on the intervals contained in η'. □

Obviously, phase type bisimulation is a fine notion of equivalence and conforms to the semantics given in terms of schedulers and path probabilities. More precisely, each scheduler decision in a PTP can be matched (stepwise) by a scheduler decision in an equivalent PTP.

Since the emphasis of this thesis is not placed on bisimulation relations we do not define weaker notions of bisimulation at this point. We claim that a subtle combination of the weak bisimulation in [Her02] and in [SL95] would yield a weak bisimulation for phase type processes.

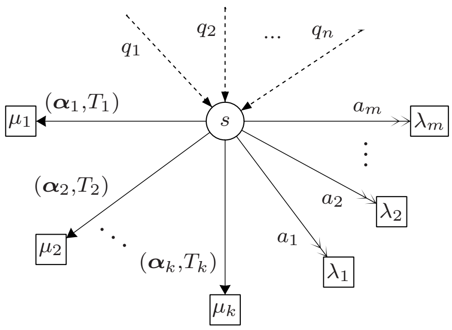

Figure 3.4: The incoming and outgoing transitions of state s

3.8 Immediate Phase Type Transitions

In this section we assume that PH transitions can have associated representations allowing a zero delay to occur with positive probability, i.e., we drop the restriction $\boldsymbol{\alpha} \cdot \mathbf{1} = 1$ for the initial distribution of a PH representation. Thus, $F_{(\boldsymbol{\alpha},T)}(0)$ might be greater than zero and a transition with label $(\boldsymbol{\alpha}, T)$ might be taken immediately. But an instantaneous state change also happens if a transition labeled by τ is taken; the difference being that for instantaneous PH transitions it is possible to specify the exact probability at which a PH transition is enabled immediately (in the case of τ-transition this probability is one). Since PH representations are required to be irreducible and to have at least one phase, we have always $F_{(\boldsymbol{\alpha},T)}(0) < 1$. Hence, it might be desirable to use both, PH transitions with general representations and τ-transitions. We gain expressiveness but it is not clear how the choice between several PH transitions and τ-transitions shall be interpreted. Let us discuss this problem in the sequel.

Assume that s is a state with $k > 0$ outgoing PH transitions $s \xrightarrow{\boldsymbol{\alpha}_i, T_i} \mu_i$, $i \in \{1, 2, \ldots, k\}$ and no other (action or PH) transitions. Then the probability of leaving s immediately equals

$$p^{(0)} := 1 - \prod_{i=1}^{k} P\big(X_i > 0\big) = 1 - \prod_{i=1}^{k}(\boldsymbol{\alpha}_i \mathbf{1}) = 1 - (\otimes_{i=1}^{k} \boldsymbol{\alpha}_i)\mathbf{1}.$$

where X_i is a PH distributed random variable with representation $(\boldsymbol{\alpha}_i, T_i)$ and $\mathbf{1}$ is a column vector of appropriate size with all entries one. Now, under the condition that s is left immediately, the choice between different PH transitions with $\boldsymbol{\alpha}_i\mathbf{1} < 1$ is then resolved with respect to the probabilities $P\big(X_i = 0\big) = 1 - \boldsymbol{\alpha}_i\mathbf{1} =: p_i^{(0)}$. More precisely, it is appropriate to assume that, under the condition that the sojourn time in s is zero, the probability of the i-th transition is given by $p_i^{(0)}/p^{(0)}$. This makes sure that PH transitions with small $p_i^{(0)}$ are chosen with a lower probability than those with a higher value $p_i^{(0)}$. E.g. if $k = 2$, $p_1^0 = 0.1$ and $p_2^0 = 0.5$, we have a zero delay with probability $p^{(0)} = 1 - (0.9 \cdot 0.5) = 0.55$. Obviously, it is much more likely that transition $s \xrightarrow{\boldsymbol{\alpha}_2, T_2} \mu_2$ triggers a zero delay in s than transition $s \xrightarrow{\boldsymbol{\alpha}_1, T_1} \mu_1$. The former happens with probability $p_2^0/p^{(0)} = 0.5/0.55 \approx 0.9$ whereas in ca. 10% of the cases the residence time in s is zero because transition $s \xrightarrow{\boldsymbol{\alpha}_1, T_1} \mu_1$ "won the race".

If s has also one or more outgoing τ-transitions, we do not have appropriate "weights" for them. All τ-transitions should have the same probability to be taken. We can assume that the scheduler determines the probability at which each τ-transition is taken under the condition that s is left immediately. But this contradicts our intuition that the weight of this transition equals the probability that this transition is enabled immediately (which is one in the case of a τ-transition) and it is not clear how to combine the scheduler decision and the weights $p_i^{(0)}$.

In the light of these problems, the restriction $\boldsymbol{\alpha} \cdot \mathbf{1} = 1$ made in the remaining parts of this paper appears most reasonable. But there is another argument for this restriction: We can simulate the case where a PH transition has representation $(\boldsymbol{\alpha}, T)$, $\boldsymbol{\alpha} \cdot \mathbf{1} < 1$ with our restricted model. Assume that s is a state with $k > 0$ outgoing PH transitions. Additionally, we assume that s has m action transitions labeled by $a_1, a_2, \ldots, a_m$ and that s is in the support of n distributions[6] with probability $q_1, q_2, \ldots, q_n$ (compare Figure 3.4).

[6]The case where s is in the support of infinitely many distributions is treated in a similar way.

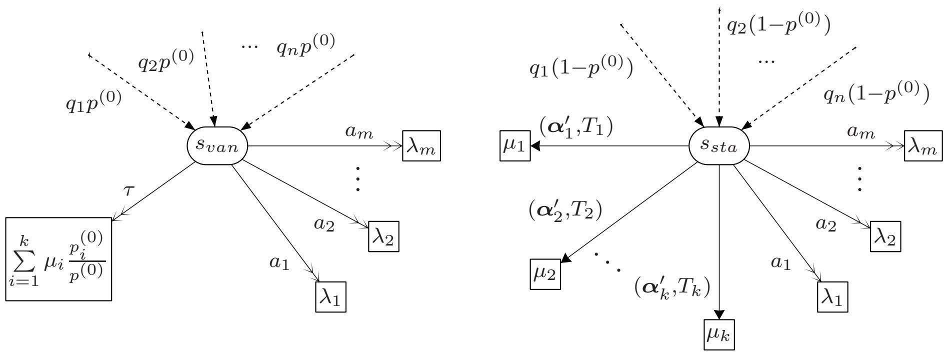

Figure 3.5: The incoming and outgoing transitions of state $s^{(i)}$ and $s^{(d)}$

The idea is to replace s by two states s_{van} and s_{sta} ("vanishing" and "stable"). The former one is left immediately because of one outgoing τ-transition with target distribution

$$\mu := \sum_{i=1}^{k} \frac{p_i^{(0)}}{p^{(0)}} \, \mu_i .$$

Moreover, s_{van} has the same action transitions as s but is reached with probability $q_1 \cdot p^{(0)}, q_2 \cdot p^{(0)}, \ldots, q_n \cdot p^{(0)}$ (compare Figure 3.5, left). The incoming edges of state s_{sta} are labeled by $q_1 \cdot (1 - p^{(0)}), q_2 \cdot (1 - p^{(0)}), \ldots, q_n \cdot (1 - p^{(0)})$ and the PH transitions have labels $(\boldsymbol{\alpha}'_i, T_i)$, $1 \leq i \leq k$ where $\alpha'_i(j) := \frac{\alpha_i(j)}{\boldsymbol{\alpha}_i \mathbf{1}}$. Note that this implies $\boldsymbol{\alpha}'_i \mathbf{1} = 1$ and therefore the residence time in s_{sta} cannot be zero. As in the case of s_{van} the action transitions emerging from s also emerge from s_{sta} (compare Figure 3.5, right).

Intuitively, the replacement of s by s_{van} and s_{sta} yields a process that is "equivalent" to the original one. Obviously, bisimulation equivalence is too fine but we claim that all linear time relations defined in the subsequent chapters identify the two processes. However, we omit a further analysis in the sequel since a proof would require new definitions of paths, schedulers, etc. for the extended model.

3.9 Chapter Summary

In this section we have presented the concept of phase type processes which is the central formalism of the thesis. Phase type processes form a very general class of models including e.g. the class of labeled transition systems, probabilistic automata and interactive Markov chains. The main difference lies in the use of phase type transitions which are enabled after a phase type distributed delay. We made use of matrix operations based on the Kronecker product to give formal semantics in terms of path probabilities. More precisely, the generator matrix of the Markov process underlying a PTP can be analytically represented using Kronecker algebra. It is important to point out that this Kronecker representation is advantageous for the model solution since the complete matrix does not need to be generated. A lot of Kronecker-based methods exist in the field of stochastic automata networks [Pla84] which exploit the structured representation to solve large Markov models [FPS98, BCDK00]. We claim that for the analysis of phase type processes it is possible to make use of the Kronecker representation as well.

We have used schedulers to resolve nondeterminism in PTPs and stuck to the classical scheduler types in [Put94]. All schedulers decide that nondeterministic alternatives are either taken immediately or completely blocked with a certain probability. They do not have the possibility to specify a time duration after which a transition is enabled. But if we view phase type processes as concurrent processes which act in continuous time, such an extended scheduler approach is suitable as well. We pick up this idea in Chapter 6 where we describe process environments in which actions are externally available after a certain amount of time.

We have defined bisimulation equivalence for PTPs which is a very fine notion of equivalence. Here, "fine" means that only few processes are related by this equivalence. A lot of variants of bisimulation equivalence exist in literature (e.g. compare [HJ89, LS91, SL95, vGSST90]). Many of them are "coarser" and can be adapted to our formalism but since the emphasis of this thesis are linear time relations we leave the investigation of coarser bisimulations as future work.

Another starting point for future research is the development of checking algorithms for our bisimulation. We claim that it is possible to extend the existing partitioning/splitter technique for probabilistic processes with non-determinism [Bai98] which runs in time polynomial in the size of the state transition graph and add a method to check the PH distributions of two given representations for equality. Proposition 2.1 (on page 27) shows that this requires time polynomial in the number of phases. More precisely, given two representations of order n and m, the comparison of the first $2 \cdot \max\{n, m\}$ moments is necessary to decide whether they describe the same distribution or not. However, the first k moments of a PH distribution with representation of order n can be computed in time $O(n^3 + k \cdot n)$ (compare Remark 2.1 on page 22). Therefore, we claim that bisimulation equivalence for PTPs can be checked in polynomial time.

Chapter 4

Parallel Composition

4.1 Overview

Many real-world systems can be considered as a set of concurrent processes, such as molecules in a living cell, Java threads, network protocols, etc. Compositional modeling affords several benefits including the possibility to describe components of a global system by first specifying subsystems and then combining the submodels by *parallel composition* to a global one.
Different forms of parallel composition exist and most of them have been analyzed in the context of process algebra [Mil80, BHR84]. Here, we use multi-way synchronization which is based on CCS [BHR84] and allow a number of processes to participate in a common transition. All synchronization partners are treated equally, i.e., there is no distinction between active and passive synchronization partners.
The parallel composition of models based on continuous time Markov chains has always been subject to keen discussions. Most stochastic process algebras follow the idea of durational actions, i.e., the execution of an action lasts an exponentially distributed amount of time. This is, for instance, the case for the stochastic process algebras TIPP [GHR92], PEPA [Hil96] and EMPA [BG96]. If now a common transition is performed, the "slowest" communication partner determines the amount of time the synchronous action lasts. Unfortunately, the maximum of two random variables following exponential distributions is not exponentially distributed. Therefore, a syn-

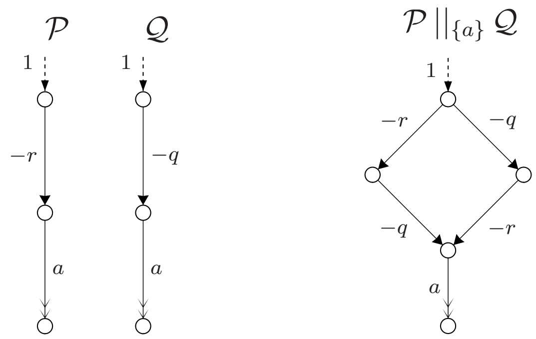

Figure 4.1: Interleaving of single phased transitions

chronous transition cannot be encoded as one transition in the compound model. This problem is circumvented by assuming that the duration of the common action is determined by only one participant, i.e., there is exactly one "active" communication partner and all others are "passive". The stochastic process algebra PEPA additionally allows an approximation of the actual duration by the calculation of a rate which involves the rates of all participants. For details on this discussion we refer to [Hil94].

Hermanns proposes a more general and very elegant solution to this problem in [Her02]. He defines a stochastic process algebra based on IMCs with a clear separation between delays and actions and sticks to the original parallel composition operator for LTSs. More precisely, processes synchronize on action transitions and timed transitions are composed in an interleaving style. The result is exactly what applies to many natural systems: the cooperation is completed when the slowest participant has finished his operation. Let us illustrate the effect of interleaved exponential delays by an example. In Figure 4.1, the two IMCs $\mathcal{P}$ and $\mathcal{Q}$ are composed such that timed transitions are interleaved and synchronization applies to a-transitions (the compound process is denoted by $\mathcal{P}\,||_{\{a\}}\,\mathcal{Q}$). The result exhibits the typical "diamond structure". From the construction of a PH representation describing the maximum of two PH distributions (see Definition 2.6 on page 25), we know that the diamond represents the distribution $F_{\max} = F_{-r} \cdot F_{-q}$.

However, when (arbitrary) phase type processes are composed via an (asynchronous) parallel composition operator, the result is not necessarily a phase type process. This is due to the absence of the memoryless property. Therefore, we first define parallel composition for the subclass of single phase type processes in Section 4.2, for which interleaving semantics in the style of interactive Markov chains is possible. In Section 4.3 we define an operator which "expands" a PTP such that all phases of a state are visible. This operator allows for a detailed discussion of the parallel composition problem of PTPs in Section 4.4.

4.2 Composition of Single Phases

In this section we focus on an adaption of the parallel composition operators in [Her02] and [LS91]. Since we restrict ourselves to SPTPs all PH transitions have only a single phase which ensures the memoryless property during composition.

Definition 4.1 (Parallel composition of SPTPs)
The parallel composition $\mathcal{P}_1 \,||_A\, \mathcal{P}_2$ of two SPTPs $\mathcal{P}_1$ and $\mathcal{P}_2$ over a finite set A of visible actions is given by SPTP

$$\mathcal{P}_1 \,||_A\, \mathcal{P}_2 \quad = \quad \big(\{(s_1 \,||_A\, s_2) \mid s_1 \in S_1, s_2 \in S_2\}, \longrightarrow, \twoheadrightarrow, \nu\big)$$

where

- $\twoheadrightarrow$ is such that $(s_1 \,||_A\, s_2) \xrightarrow{a}\!\!\!\!\rightarrow \mu$ iff one of the following conditions is true:

  i) $a \in A \wedge \exists \lambda_1, \lambda_2 :$
  $s_1 \xrightarrow{a}\!\!\!\!\rightarrow_1 \lambda_1 \wedge s_2 \xrightarrow{a}\!\!\!\!\rightarrow_2 \lambda_2 \wedge \big(\forall s_1', s_2' :\ \lambda_1(s_1') \cdot \lambda_2(s_2') = \mu(s_1' \,||_A\, s_2')\big)$

  ii) $a \notin A \wedge \exists \lambda_1 :$
  $s_1 \xrightarrow{a}\!\!\!\!\rightarrow_1 \lambda_1 \wedge \big(\forall s_1' :\ \lambda_1(s_1') = \mu(s_1' \,||_A\, s_2)\big)$

  iii) $a \notin A \wedge \exists \lambda_2 :$
  $s_2 \xrightarrow{a}\!\!\!\!\rightarrow_2 \lambda_2 \wedge \big(\forall s_2' :\ \lambda_2(s_2') = \mu(s_1 \,||_A\, s_2')\big)$

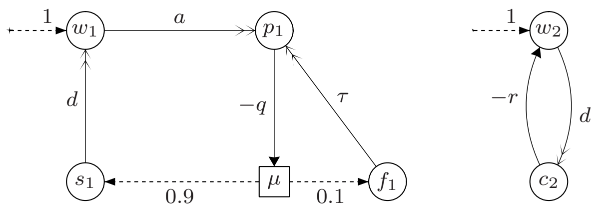

Figure 4.2: A model of the producer (left) and the consumer (right)

- $\longrightarrow$ is such that $(s_1 \,||_A\, s_2) \xrightarrow{-r} \mu, r > 0$, iff either

$$\big(s_2 \text{ is stable} \wedge \exists \lambda_1 : \; s_1 \xrightarrow{-r}{}_1 \lambda_1 \wedge \forall s_1' : \; \mu(s_1' \,||_A\, s_2) = \lambda_1(s_1')\big) \text{ or}$$
$$\big(s_1 \text{ is stable} \wedge \exists \lambda_2 : \; s_2 \xrightarrow{-r}{}_2 \lambda_2 \wedge \forall s_2'. \; \mu(s_1 \,||_A\, s_2') = \lambda_2(s_2')\big).$$

- The initial distribution ν is such that $\nu(s_1 \,||_A\, s_2) = \nu_1(s_1) \cdot \nu_2(s_2)$ for all $s_1 \in S_1, s_2 \in S_2$.

Here, set A determines which actions have to be performed synchronously and $\mathsf{Act} \setminus A$ contains the actions that are performed independently. Thus, it may happen that one process, say $\mathcal{P}$, has to wait for $\mathcal{Q}$ to perform a transition with label $a \in A$. Such a transition cannot be executed until a is also possible in the (current) local state of $\mathcal{Q}$. If the set A of common actions equals the empty set we shortly write $||$ instead of $||_\emptyset$. Note that the parallel composition of finitely branching SPTPs is a SPTP that is finitely branching as well. Similarly, if we compose two SPTPs that do not show zeno behavior, the compound process also does not have zeno paths. The same holds for divergence.

Example 4.1

Consider the two SPTPs in Figure 4.2. Their parallel composition over action set $A = \{d\}$ yields the SPTP of Example 3.1 (see also Figure 3.1 on page 37) if we assume that $(\boldsymbol{\alpha}, T) = (1, -q)$ for some $q > 0$.

We conclude this section by observing that phase type bisimulation is preserved by parallel composition of SPTPs. The proof is similar to that in [Her02] and [vGSST90] and is therefore omitted here.

Proposition 4.1
Let $\mathcal{P}_1, \mathcal{P}_2$ *be* SPTP*s with* $\mathcal{P}_1 =_{\mathsf{bs}} \mathcal{P}_2$. *Then for all* SPTP*s* $\mathcal{Q}$, $A \subseteq \mathsf{Act}$

$$\mathcal{P}_1 \,||_A\, \mathcal{Q} =_{\mathsf{bs}} \mathcal{P}_2 \,||_A\, \mathcal{Q}.$$

4.3 Expand Operator

This section discusses an operator that turns out to be useful in the context of parallel composition. It is based on the observation that a PTP constitutes a “compressed” model in which the phases of a state are hidden. Intuitively, a PTP, say $\mathcal{P}_1$, may provide a more abstract view on a system than another PTP, say $\mathcal{P}_2$, in which several phases (represented by several PH transitions) are collapsed to one PH transition. Consider for example a PTP modeling DNA transcription. The time until the transcription of a gene is finished can be expressed as a phase type distributed random variable which is the sum of several exponentially distributed random variables. If the modeler decides that the intermediate transcription steps are important he might use several single phased PH transitions to model transcription. If a more abstract view is desired a single PH transition labeled by an appropriate representation can be used.
Stated more generally, the ex-operator (from “expand”) assigns to each phase type process $\mathcal{P}$ a single phase type process $\mathcal{I} = \mathsf{ex}(\mathcal{P})$. The idea is that intermediate steps of the stochastic delays in $\mathcal{P}$ are specified explicitly.

Definition 4.2 (Expand operator)
Let $\mathcal{P}$ be a PTP and for $s, u \in S_{\mathcal{P}}$ let $n_s, \boldsymbol{\gamma}_s, Q_{s,u}$ be as in Definition 3.3 (see page 38). The SPTP $\mathsf{ex}(\mathcal{P}) =: \mathcal{I}$ is given by $S_{\mathcal{I}} := \{s^{(i)} \mid s \in S_{\mathcal{P}}, 1 \leq i \leq n_s\}$ and for all $s \in S_{\mathcal{P}}$

- $\forall i, j \in \{1, \ldots, n_s\} : s^{(i)} \xrightarrow{-r}_{\mathcal{I}} s^{(j)}$ iff $Q_{s,s}(i,j) = r > 0$ and

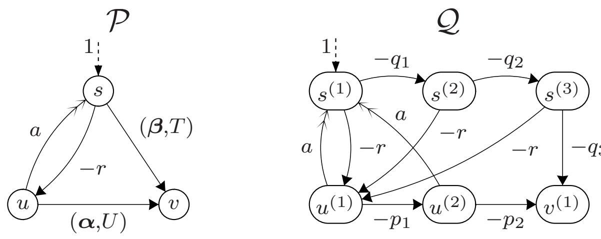

Figure 4.3: $\mathcal{P}$ and $\mathcal{Q} = \mathsf{ex}(\mathcal{P})$

$$\forall u \in S_{\mathcal{P}}, u \neq s, \forall k \in \{1, \ldots, n_u\} : s^{(i)} \xrightarrow{-r}_{\mathcal{I}} u^{(k)} \text{ iff } Q_{s,u}(i,k) = r > 0,$$

- $\forall i \in \{1, \ldots, n_s\} : s^{(i)} \xrightarrow{a}\!\!\!\rightarrow_{\mathcal{I}} \mu$ iff $s \xrightarrow{a}\!\!\!\rightarrow_{\mathcal{P}} \mu'$ and

$$\forall u \in S_{\mathcal{P}}, j \in \{1, \ldots, n_u\} : \mu'(u) \cdot \gamma_u(j) = \mu(u^{(j)}),$$

- $\forall i \in \{1, \ldots, n_s\} : \nu_{\mathcal{I}}(s^{(i)}) = \nu_{\mathcal{P}}(s) \cdot \gamma_s(i).$

Note that ex replaces the discrete target distributions of PH transitions by Dirac distributions and takes the entries of the generator matrix as rate labels. More precisely, if $(s, (\boldsymbol{\alpha}, T), \mu)$ is a PH transition in $\mathcal{P}$ then the absorption rates of vector $\mathbf{T}^0$ are distributed according to μ such that $s^{(i)} \xrightarrow{r} u^{(k)}$, $s \neq u$ which means that $r = \mathbf{T}^0(i) \cdot \mu(u) \cdot \gamma_u(k)$. This means that we multiply the absorption rate of phase i with the probability to enter the k-th phase of state u after absorption. Alternatively, one could define ex such that $s^{(i)}$ has a transition with rate $\mathbf{T}^0(i)$ leading to distribution $\mu(u) \cdot \gamma_u \in \mathsf{dis}(\{u^{(1)}, u^{(2)}, \ldots, u^{(n_u)}\})$. Note also that ex removes all PH self-loops of state s if $n_s = 1$.

Example 4.2

Consider the two PTPs in Figure 4.3. For simplicity, all target distributions are Dirac distributions. We define matrices

$$U := \begin{bmatrix} -p_1 & p_1 \\ 0 & -p_2 \end{bmatrix}, T := \begin{bmatrix} -q_1 & q_1 & 0 \\ 0 & -q_2 & q_2 \\ 0 & 0 & -q_3 \end{bmatrix}$$

and vectors

$$\boldsymbol{\alpha} := \begin{bmatrix} 1 & 0 \end{bmatrix}, \boldsymbol{\beta} := \begin{bmatrix} 1 & 0 & 0 \end{bmatrix}.$$

We assume that $r \in \mathbb{R}_{>0}$. It is easy to see that $\mathcal{Q} = \mathsf{ex}(\mathcal{P})$ by checking the conditions of Definition 4.2 and calculating

$$\begin{aligned} \boldsymbol{\gamma}_s &= \begin{bmatrix} 1 & 0 & 0 \end{bmatrix}, \\ Q_{s,s} &= \begin{bmatrix} -(r+q_1) & q_1 & 0 \\ 0 & -(r+q_2) & q_2 \\ 0 & 0 & -(r+q_3) \end{bmatrix}, \\ Q_{s,v} &= \begin{bmatrix} 0 \\ 0 \\ q_3 \end{bmatrix}, \quad Q_{s,u} = \begin{bmatrix} r \\ r \\ r \end{bmatrix}. \end{aligned}$$

Next we focus on the relationship between $\mathcal{P}$ and $\mathsf{ex}(\mathcal{P})$. Intuitively, one might argue that they are "equivalent" in some sense. However, they are *not* related by phase type bisimulation (see Definition 3.7) since those states of $\mathsf{ex}(\mathcal{P})$ that represent phases of a state in $\mathcal{P}$ may not be related to that state. On the other hand, $\mathcal{P}$ and $\mathsf{ex}(\mathcal{P})$ have the same generator matrix (up to enumeration of the state space) which follows directly from construction. Therefore, they have the same underlying continuous time Markov chain. Moreover, the states in $\mathsf{ex}(\mathcal{P})$ representing the phases of a state in $\mathcal{P}$ inherit its action transitions. In the sequel, we show that this results in "equivalent" paths probabilities. To formalize this we first aim at the relationship between path fragments of $\mathcal{P}$ and $\mathsf{ex}(\mathcal{P})$. For the remainder of this section, assume that there is no state that is in the support of a discrete distribution being the target of a PH transition emerging from that state (PH self-loops). This restriction simplifies the distinction between the case that the process visits different phases of a state and the case where a PH self-loop is taken. We claim that the following results can also be proven for the general case but the proofs would require some technical effort in order to mark the execution of a self-loop of $\mathcal{P}$ in a path of $\mathsf{ex}(\mathcal{P})$.

Let $\xi \in \mathsf{pathf}(\mathsf{ex}(\mathcal{P}))$. ξ is called $\mathcal{P}$-*observable* if for all states s of $\mathcal{P}$ each maximal subsequence

$$s^{(i_1)} \xrightarrow{t_1} s^{(i_2)} \xrightarrow{t_2} \dots \xrightarrow{t_{m-1}} s^{(i_m)}, \quad m > 1$$

of ξ is followed by $s^{(i_m)} \xrightarrow{t_m} u^{(k)}, u \neq s$ where $t_j \in \mathbb{R}_{>0}$, $i_j \in \{1, 2, \dots, n_s\}$ for all $j \in \{1, 2, \dots, m\}$. If ξ is $\mathcal{P}$-observable we write $\mathsf{contr}(\xi)$ (from "contraction") for the sequence that results from ξ by replacing maximal subsequences

$$s^{(i_1)} \xrightarrow{t_1} s^{(i_2)} \xrightarrow{t_2} \dots \xrightarrow{t_{m-1}} s^{(i_m)} \xrightarrow{t_m} u^{(k)}, u \neq s, m > 1$$

by $s \xrightarrow{t} u$, $t = \sum_{j=1}^{m} t_j$ and $v^{(i)} \xrightarrow{a} w^{(l)}$ by $v \xrightarrow{a} w$ for all $a \in \mathsf{Act}_\tau$, $i \in \{1, \dots, n_v\}$, $l \in \{1, \dots, n_w\}$ and all states s, u, v, w in $\mathcal{P}$.

Example 4.3

Recall Example 4.2. Instances of $\mathcal{P}$-observable path fragments of $\mathcal{Q} = \mathsf{ex}(\mathcal{P})$ (see Figure 4.3) are

$$\begin{aligned} \xi_1 &= s^{(1)} \xrightarrow{0.2} s^{(2)} \xrightarrow{0.3} u^{(1)} \xrightarrow{a} s^{(1)}, \\ \xi_2 &= s^{(1)} \xrightarrow{0.2} s^{(2)} \xrightarrow{0.5} s^{(3)} \xrightarrow{0.1} v^{(1)} \end{aligned}$$

with $\mathsf{contr}(\xi_1) = s \xrightarrow{0.5} u \xrightarrow{a} s$ and $\mathsf{contr}(\xi_2) = s \xrightarrow{0.8} v$. An instance of a path fragment that is not $\mathcal{P}$-observable is

$$s^{(1)} \xrightarrow{0.2} u^{(1)} \xrightarrow{0.3} u^{(2)} \xrightarrow{a} s^{(1)}.$$

Remark 4.1

Note that it is possible to define a scheduler such that all path fragments of $\mathsf{ex}(\mathcal{P})$ *are observable in* $\mathcal{P}$. *More precisely, if we consider a scheduler more as* *an instance simulating the environment of a* PTP *rather than a strategy for instantaneous decisions, it is appropriate to define a scheduler that decides for a visible action after a certain time period has passed (compare also Remark 3.1 on page 49). This idea is picked up in Section 6.3.*

It is easy to see that $\mathsf{contr}(\xi) \in \mathsf{pathf}(\mathcal{P})$. Moreover, mapping contr is surjective. Now assume that $\mathcal{D}$ is a THR-scheduler for $\mathsf{ex}(\mathcal{P})$. We call $\mathcal{D}$ a $\mathcal{P}$-*observation-based* scheduler if the following two conditions hold

- whenever ξ_1 and ξ_2 are $\mathcal{P}$-observable and $\mathsf{contr}(\xi_1) = \mathsf{contr}(\xi_2)$ then $\mathcal{D}(\xi_1) = \mathcal{D}(\xi_2)$,
- whenever ξ is not a prefix of some $\mathcal{P}$-observable path fragment then it holds that $\mathsf{Pr}^{\mathcal{D}}(\xi) = 0$.

These conditions ensure that $\mathcal{D}$'s decisions depend only on the history in $\mathcal{P}$ and path fragments with a non-zero probability must be completed to $\mathcal{P}$-observable ones. It is easy to see that this implies that a $\mathcal{P}$-observation-based scheduler chooses $\mathcal{D}(\xi) = \lambda$ with $\lambda^{\perp} = 1$ if ξ is not $\mathcal{P}$-observable but a prefix of a $\mathcal{P}$-observable path fragment.

Example 4.4
Recall Example 4.2. Path fragment

$$\xi = s^{(1)} \xrightarrow{0.2} u^{(1)} \xrightarrow{0.3} u^{(2)}$$

is not $\mathcal{P}$-observable but a prefix of a $\mathcal{P}$-observable path fragment. If scheduler $\mathcal{D} \in \mathsf{THR}(\mathcal{P})$ chooses $\mathcal{D}(\xi)^{\perp} = 1$ all continuations of ξ that are not $\mathcal{P}$-observable have probability zero.

Let $\zeta = s_1\ E_1\ s_2\ E_2 \ldots\ E_{k-1}\ s_k$ where $s_1, s_2, \ldots, s_k \in S_{\mathcal{P}}$ and for $i \in \{1, 2, \ldots, k-1\}$ either $E_i = \{a\}$ for some $a \in \mathsf{Act}_\tau$ or $E_i = (x, y] \subseteq \mathbb{R}_{>0}$, $x < y$. Let cylinder set C_ζ be the set of all paths $\pi \in \mathsf{path}(\mathcal{P})$ such that

$$\pi{\downarrow_k} = s_1 \xrightarrow{\mathsf{e}_1} s_2 \xrightarrow{\mathsf{e}_2} \ldots \xrightarrow{\mathsf{e}_{k-1}} s_k$$

and $\mathsf{e}_1 \in E_i$ for all $i \in \{1, 2, \ldots, k-1\}$. Let $\mathcal{C}_\mathcal{P}$ be the set of all such cylinder sets. For $\mathsf{C}_\zeta \in \mathcal{C}_\mathcal{P}$ we define $\mathsf{ex}(\mathsf{C}_\zeta)$ as the set of all paths $\pi \in \mathsf{path}(\mathsf{ex}(\mathcal{P}))$ for which there exists $j \geq k$ such that $\pi{\downarrow}_j$ is $\mathcal{P}$-observable and

$$\mathsf{contr}(\pi{\downarrow}_j) = s_1 \xrightarrow{\mathsf{e}_1} s_2 \xrightarrow{\mathsf{e}_2} \ldots \xrightarrow{\mathsf{e}_{k-1}} s_k$$

is a prefix of a path in C_ζ. Hence, if $\zeta = s$ then $\mathsf{ex}(\mathsf{C}_s)$ is the set of all paths in $\mathsf{ex}(\mathcal{P})$ that start in a state $s^{(i)}$ for some $i \in \{1, \ldots, n_s\}$.

Theorem 4.1

Let $\mathcal{P}$ be a PTP. Then for each $\mathcal{P}$-observation-based scheduler $\mathcal{D} \in \mathsf{THR}(\mathsf{ex}(\mathcal{P}))$ there exists a scheduler $\mathcal{D}' \in \mathsf{THR}(\mathcal{P})$ such that for all $\mathsf{C}_\zeta \in \mathcal{C}_\mathcal{P}$

$$\mathsf{Pr}^{\mathcal{D}}_{\mathsf{ex}(\mathcal{P})}(\mathsf{ex}(\mathsf{C}_\zeta)) = \mathsf{Pr}^{\mathcal{D}'}_{\mathcal{P}}(\mathsf{C}_\zeta),$$

and vice versa.

Proof. Let us first prove the existence of $\mathcal{D}'$ if $\mathcal{D} \in \mathsf{THR}(\mathsf{ex}(\mathcal{P}))$ is given. We use the fact that contr is surjective and define $\mathcal{D}'(\xi')(a, \mu') := \mathcal{D}(\xi)(a, \mu)$ for $\xi' \in \mathsf{pathf}(\mathcal{P})$ where $\xi \in \mathsf{pathf}(\mathsf{ex}(\mathcal{P}))$ is such that $\xi' = \mathsf{contr}(\xi)$ and $\mu'(u) \cdot \gamma_u(j) = \mu(u^{(j)})$ for all $u \in S_\mathcal{P}$, $j \in \{1, \ldots, n_u\}$ (compare Definition 4.2). Since $\mathcal{D}$ is $\mathcal{P}$-observation-based, $\mathcal{D}'(\xi')$ is well-defined (compare the first condition of the definition of a $\mathcal{P}$-observation-based scheduler). The next step is to prove by induction on the length of ζ that

$$\mathsf{Pr}^{\mathcal{D}}_{\mathsf{ex}(\mathcal{P})}(\mathsf{ex}(\mathsf{C}_\zeta)) = \mathsf{Pr}^{\mathcal{D}'}_{\mathcal{P}}(\mathsf{C}_\zeta) \tag{4.1}$$

where ζ is such that the decisions of $\mathcal{D}'$ (and therefore those of $\mathcal{D}$) are constant on the intervals contained in ζ. But having established (4.1) the statement follows immediately because we can express any set in $\mathcal{C}_\mathcal{P}$ as the disjoint union of sets C_ζ for which $\mathcal{D}$ and $\mathcal{D}'$ have constant choices on the intervals of ζ. Assume that $\mathsf{C}_\zeta = C_s$. Then

$$\begin{aligned}
\mathsf{Pr}^{\mathcal{D}}_{\mathsf{ex}(\mathcal{P})}(\mathsf{ex}(\mathsf{C}_s)) &= \mathsf{Pr}^{\mathcal{D}}_{\mathsf{ex}(\mathcal{P})}\{\pi \in \mathsf{path}(\mathsf{ex}(\mathcal{P})) \mid \\
&\qquad \pi{\downarrow_1} = s^{(i)} \text{ for some } i \in \{1,\ldots,n_s\}\} \\
&= \textstyle\sum_{i=1}^{n_s} \nu(s)\gamma_s(i) \\
&= \nu(s) \\
&= \mathsf{Pr}^{\mathcal{D}'}_{\mathcal{P}}(\mathsf{C}_s).
\end{aligned}$$

For the induction step, let

$$\zeta' = s_1\ E_1\ s_2\ E_2 \ldots E_{k-1}\ s_k \text{ and } \zeta = s_1\ E_1\ s_2\ E_2 \ldots E_{k-2}\ s_{k-1}.$$

Let ξ be a $\mathcal{P}$-observable path fragment in $\mathsf{ex}(\mathcal{P})$ such that $\mathsf{contr}(\xi) = \pi{\downarrow_{k-1}}$ with $\pi \in \mathsf{C}_\zeta$. The choice $\mathcal{D}(\xi)$ depends only on ζ since $\mathcal{D}$ is a $\mathcal{P}$-observation-based scheduler and has constant choices on the intervals of ζ. Therefore we write $D(\zeta)$ instead of $D(\xi)$. The same holds for $\mathcal{D}'(\pi{\downarrow_{k-1}})$. Let $\lambda := D(\zeta)$ and $\mathcal{D}'(\zeta) = \lambda'$. If $E_{k-1} = \{a\}$, $a \in \mathsf{Act}_\tau$, we derive that $\mathsf{Pr}^{\mathcal{D}}_{\mathsf{ex}(\mathcal{P})}(\mathsf{ex}(\mathsf{C}_{\zeta'}))$ equals

$$\mathsf{Pr}^{\mathcal{D}}_{\mathsf{ex}(\mathcal{P})}(\mathsf{ex}(\mathsf{C}_\zeta)) \cdot \left\{\!\!\left| \sum_{l=1}^{n_{s_k}} \lambda(a,\mu)\cdot\mu(s_k^{(l)}) \mid s_{k-1} \xrightarrow{a}\!\!\!\!\rightarrow \mu', \forall s^{(j)} : \mu'(s)\gamma_s(j) = \mu(s_k^{(j)}) \right|\!\!\right\}$$

by using the fact that all states $s_{k-1}^{(i)}$, $i \in \{1,\ldots,n_{s_{k-1}}\}$ inherit the a-transitions of s_{k-1} and that $\mathcal{D}$ is a $\mathcal{P}$-observation-based scheduler. From $\lambda(a,\mu) = \lambda'(a,\mu')$ we get

$$\begin{aligned}
& \left\{\!\!\left| \textstyle\sum_{l=1}^{n_{s_k}} \lambda(a,\mu)\cdot\mu(s_k^{(l)}) \mid s_{k-1} \xrightarrow{a}\!\!\!\!\rightarrow \mu', \forall s,j : \mu'(s)\gamma_s(j) = \mu(s^{(j)}) \right|\!\!\right\} \\
= & \left\{\!\!\left| \textstyle\sum_{l=1}^{n_{s_k}} \lambda'(a,\mu')\cdot\mu'(s_k)\gamma_{s_k}(l) \mid s_{k-1} \xrightarrow{a}\!\!\!\!\rightarrow \mu' \right|\!\!\right\} \\
= & \left\{\!\!\left| \lambda'(a,\mu')\cdot\mu'(s_k) \mid s_{k-1} \xrightarrow{a}\!\!\!\!\rightarrow \mu' \right|\!\!\right\}
\end{aligned}$$

which implies that

$$\begin{aligned}
\mathsf{Pr}^{\mathcal{D}}_{\mathsf{ex}(\mathcal{P})}(\mathsf{ex}(\mathsf{C}_{\zeta'})) &= \mathsf{Pr}^{\mathcal{D}}_{\mathsf{ex}(\mathcal{P})}(\mathsf{ex}(\mathsf{C}_{\zeta})) \cdot \Big\{\!\!\Big| \textstyle\sum_{l=1}^{n_{s_k}} \lambda(a,\mu)\cdot\mu(s_k^{(l)}) \mid \\
&\qquad s_{k-1} \xrightarrow{a} \mu', \forall s^{(j)} : \mu'(s)\gamma_s(j) = \mu(s_k^{(j)}) \Big|\!\!\Big\} \\
&= \mathsf{Pr}^{\mathcal{D}}_{\mathsf{ex}(\mathcal{P})}(\mathsf{ex}(\mathsf{C}_{\zeta})) \cdot \Big\{\!\!\Big| \lambda'(a,\mu')\cdot\mu'(s_k) \mid s_{k-1} \xrightarrow{a} \mu' \Big|\!\!\Big\} \\
&\overset{\text{ind. hyp.}}{=} \mathsf{Pr}^{\mathcal{D}'}_{\mathcal{P}}(\mathsf{C}_{\zeta}) \cdot \Big\{\!\!\Big| \lambda'(a,\mu')\cdot\mu'(s_k) \mid s_{k-1} \xrightarrow{a} \mu' \Big|\!\!\Big\} \\
&= \mathsf{Pr}^{\mathcal{D}'}_{\mathcal{P}}(\mathsf{C}_{\zeta'})
\end{aligned}$$

and the proof is complete for the case that $E_{k-1} = \{a\}$.

If E_{k-1} is an interval then

$$\begin{aligned}
\mathsf{Pr}^{\mathcal{D}}_{\mathsf{ex}(\mathcal{P})}(\mathsf{ex}(\mathsf{C}_{\zeta'})) &= \mathsf{Pr}^{\mathcal{D}}_{\mathsf{ex}(\mathcal{P})}(\mathsf{ex}(\mathsf{C}_{\zeta})) \cdot \mathcal{D}(\zeta)^{\perp} \cdot \textstyle\sum_{l=1}^{n_{s_{k-1}}} \gamma_{s_{k-1}}(l) \cdot \\
&\quad \mathsf{reach}_{s_{k-1}^{(l)}}(\{s_{k-1}^{(j)} \mid 1 \le j \le n_{s_{k-1}}\}, \\
&\qquad \{s_k^{(i)} \mid 1 \le i \le n_{s_k}\}, E_{k-1}).
\end{aligned}$$

Now, recall the definition of $\mathsf{reach}_s(\{s\},\{u\},J)$ for states s, u of a PTP and an interval J (see Equation 3.1 on page 43). We consider the Markov chain underlying all PH transitions of s and start in state s with probability $\nu(s) = 1$. But then $\nu_{\{s\}} = \nu(s)\cdot\gamma_s = \gamma_s$ which means that the initial phase of s is chosen by γ_s. Combining this with the fact that $\mathcal{P}$ and $\mathsf{ex}(\mathcal{P})$ have the same generator matrix we derive

$$\begin{aligned}
&\sum_{l=1}^{n_{s_{k-1}}} \gamma_{s_{k-1}}(l) \cdot \mathsf{reach}_{s_{k-1}^{(l)}}(\{s_{k-1}^{(j)} \mid 1 \le j \le n_{s_{k-1}}\}, \{s_k^{(i)} \mid 1 \le i \le n_{s_k}\}, E_{k-1}) \\
&= \mathsf{reach}_{s_{k-1}}(\{s_{k-1}\},\{s_k\},E_{k-1}).
\end{aligned}$$

By using $\mathcal{D}(\zeta)^{\perp} = \mathcal{D}'(\zeta)^{\perp}$ we conclude that

$$\begin{aligned}
\mathsf{Pr}^{\mathcal{D}}_{\mathsf{ex}(\mathcal{P})}(\mathsf{ex}(\mathsf{C}_{\zeta'})) &= \mathsf{Pr}^{\mathcal{D}}_{\mathsf{ex}(\mathcal{P})}(\mathsf{ex}(\mathsf{C}_{\zeta})) \cdot \mathcal{D}(\zeta)^{\perp} \cdot \textstyle\sum_{l=1}^{n_{s_{k-1}}} \gamma_{s_{k-1}}(l) \cdot \\
&\quad \mathsf{reach}_{s_{k-1}^{(l)}}(\{s_{k-1}^{(j)} \mid 1 \leq j \leq n_{s_{k-1}}\}, \\
&\qquad \{s_k^{(i)} \mid 1 \leq i \leq n_{s_k}\}, E_{k-1}) \\
&= \mathsf{Pr}^{\mathcal{D}}_{\mathsf{ex}(\mathcal{P})}(\mathsf{ex}(\mathsf{C}_{\zeta})) \cdot \mathcal{D}'(\zeta)^{\perp} \cdot \mathsf{reach}_{s_{k-1}}(\{s_{k-1}\}, \{s_k\}, E_{k-1}) \\
&\overset{\text{ind. hyp.}}{=} \mathsf{Pr}^{\mathcal{D}}_{\mathcal{P}}(\mathsf{C}_{\zeta}) \cdot \mathcal{D}'(\zeta)^{\perp} \cdot \mathsf{reach}_{s_{k-1}}(\{s_{k-1}\}, \{s_k\}, E_{k-1}) \\
&= \mathsf{Pr}^{\mathcal{D}'}_{\mathcal{P}}(\mathsf{C}_{\zeta'})
\end{aligned}$$

For the reverse direction we set $\mathcal{D}(\xi)(a, \mu) := \mathcal{D}'(\xi', \mu')$ if ξ is $\mathcal{P}$-observable, $\mathsf{contr}(\xi) = \xi'$ and $\mu'(u) \cdot \gamma_u(j) = \mu(u^{(j)})$ for all u, $j \in \{1, \ldots, n_u\}$. If ξ is not $\mathcal{P}$-observable but a prefix of a $\mathcal{P}$-observable path fragment, we let $\mathcal{D}(\xi) := \lambda$ with $\lambda^{\perp} = 1$. The remaining choices of $\mathcal{D}$ are arbitrary since they are based on a history which is not $\mathcal{P}$-observable. The decisions specified so far are such that path fragments which cannot be continued to $\mathcal{P}$-observable ones have probability zero. From construction $\mathcal{D}$ is a $\mathcal{P}$-observation-based scheduler. Similar as above the statement can now be proven by induction on the length of C_{ζ}. □

Note that the above proposition indicates that whenever we consider $\mathsf{ex}(\mathcal{P})$ instead of $\mathcal{P}$ we have to restrict to $\mathcal{P}$-observable schedulers.

4.4 Composition with Partial Memory

Let us now consider the parallel composition of arbitrary PTPs $\mathcal{P}$ and $\mathcal{Q}$. Assume that $\mathcal{P}$ is waiting for synchronization on some $a \in A$. Simultaneously, the race between the PH transitions of $\mathcal{P}$'s current local state takes place and if the delay of a certain PH transition is over it is taken unless a is unblocked before (for example, because it is offered by a communication partner). This

means that, while $\mathcal{P}$ passes through the phases of its local state, say s, the action transitions of s can become enabled before absorption takes place.
The following example shows that the parallel composition of PTPs cannot be described in a satisfying way by a (global) PTP (without applying ex) even if they do not synchronize on any action.

Example 4.5
First, recall that we omit the initial distribution $\boldsymbol{\alpha}$ of a PH representation if probability one is assigned to the first state. Consider the PTPs $\mathcal{P}$ and $\mathcal{Q}$ in Figure 4.4 (left) where the representations U and V are given by

$$U = \begin{bmatrix} -r_1 & r_1 \\ 0 & -r_2 \end{bmatrix}, \quad V = \begin{bmatrix} -q_1 & q_1 \\ 0 & -q_2 \end{bmatrix}$$

and $r_1, r_2, q_1, q_2 > 0$. Figure 4.4 (right) illustrates the parallel composition of the two SPTPs $\mathsf{ex}(\mathcal{P})$ and $\mathsf{ex}(\mathcal{Q})$ over action set $A = \emptyset$. We use the ex-operator at this point to gain insight into the behavior of the parallel composition of $\mathcal{P}$ and $\mathcal{Q}$. The structure of $\mathsf{ex}(\mathcal{P}) \,||\, \mathsf{ex}(\mathcal{Q})$ is similar to the race structure of $U \otimes V$ but in case, say, U wins the race, representation V proceeds with its *current* phase (and not with the initial one). Consider, for example, the state represented as a black node in $\mathsf{ex}(\mathcal{P}) \,||\, \mathsf{ex}(\mathcal{Q})$. PTP $\mathcal{P}$ has already finished its delay and is ready to perform a while $\mathcal{Q}$ has only finished one of the two successive phases. Such intermediate states cannot be part of a PTP that models the compound process *without* an explicit reference to the phases of the states of $\mathcal{Q}$ and $\mathcal{P}$. The problem is that we have to store in which phases the states of $\mathcal{Q}$ and $\mathcal{P}$ currently are.
It is important to point out that PTP $\mathcal{P}'$ (Figure 4.4, middle) does *not* show the behavior that we expect from the parallel composition of $\mathcal{P}$ and $\mathcal{Q}$. To see this, consider again the black node. If in the initial state the left PH transition wins the race, the delay of the transition labeled by V is no longer distributed according to V. This comes from the fact that PH distributions possess only the partial memoryless property, i.e., the Markov chain that corresponds to representation V might be already in the second phase. Thus,

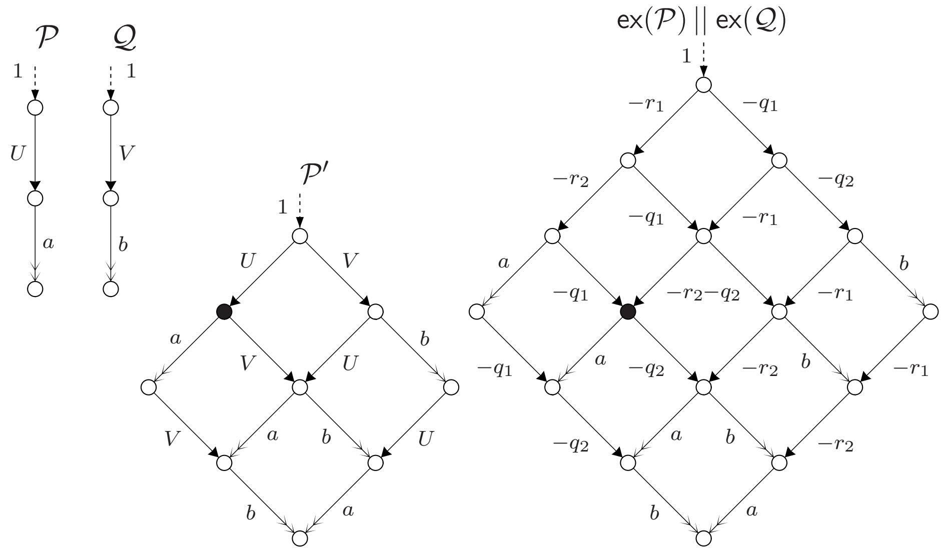

Figure 4.4: Parallel composition of the expanded PTPs

the usual interleaving semantics is not appropriate in the case of phase type processes.

4.5 Chapter Summary

This chapter has focused on the parallel composition. In the style of [Her02], an operator for SPTPs has been defined which maintains the usual interleaving semantics. It has been shown that this is not possible in the case of states having several phases. More precisely, there is, in general, no "natural" operator for the parallel composition of PTPs.

The ex-operator provides the possibility to consider a PTP on a less abstract level. The phases each state has to pass through until it is left are added as extra states. The operator is useful in many respects because it sharpens the understanding of several problems related to phase type processes, e.g. the parallel composition or the definition of certain scheduler classes. In the remaining chapters of this thesis, notions of equivalence will be defined for PTPs and we will check if ex is compatible with each of these notions.

We conclude this chapter by observing that a way out of the problem of the parallel composition of phase type processes is the use of the ex-operator and proceeding with the "expanded" model. However, in this case we would loose the advantages phase type processes have over single phase type processes. However, it is possible to define an operator that transforms a SPTP into a PTP having less states by combining several successive PH transitions to a single one.

Chapter 5

Trace Semantics

5.1 Overview

This chapter covers *trace semantics* for phase type processes. The basic idea is to explore the different execution paths a PTP can follow which means that the observable process behavior is analyzed from a *linear-time perspective* [vGla90]. For a fixed path each moment in time has a unique possible future. Thus, linear-time semantics can be captured by (linear) sequences and we regard them as describing the behavior of a single run. If the system operates under the control of a scheduler these sequences occur with a certain probability. Moreover, a PTP acts in continuous time which means that the timing behavior is also recorded. Since the observable part of a path is given by its trace, in our setting observations are functions, each of which assigns a probability to the input pair consisting of a trace and a time bound.

Trace semantics induces an equivalence relation[1], called *trace equivalence*, as follows: Two PTPs $\mathcal{P}$ and $\mathcal{Q}$ are distinguished if they can be differentiated by their observable behavior, or, in other words, $\mathcal{P}$ and $\mathcal{Q}$ are related if they have the same observation functions.

We characterize trace semantics in terms of intuitive *button pushing experiments* [Mil89, vGla90] and extend van Glabbeek's *trace machine* [vGla90] to

[1]We focus on equivalences rather than preorders. Obviously, each semantics also induces a preorder.

the stochastic setting. The trace machine of a PTP $\mathcal{P}$ is a black box which simulates $\mathcal{P}$. An observer watching the box can see those activities of $\mathcal{P}$ that are *external*. This means that they are visible to the process' environment (e.g. other "objects" which might interact with $\mathcal{P}$). Visible actions are shown at an *action display*. Since PTPs act in continuous-time, we put the observer in the position to clock how long the execution of a trace takes. This is realized with an *hourglass (timer)*. We call the machine extended in this way *stochastic trace machine*. An illustration is given in Figure 5.1 on page 85.

Several notions of trace equivalence are defined which differ in the type of scheduler used to resolve nondeterministic choices. In the case of history-dependent schedulers, the resulting relations can be viewed as continuous-time counterparts of Segala's notion of trace distribution equivalence for probabilistic automata [Seg95]. Surprisingly, the choice of the scheduler-type is crucial, as the induced trace equivalences are different, and even not comparable in most cases. Similar results have been established in [WBM06].

We are also concerned with variants of trace semantics, namely semantics based on completed traces [BW82], failures [BHR84] and ready sets [OH83]. In the case of failures, we define the *stochastic failure machine* (see Figure 5.6 on page 108) by equipping the stochastic trace machine with switches which are used to block or unblock actions. Similarly, for ready semantics, (action) lamps are used to reveal which actions are currently externally enabled (see Figure 5.7 on page 110). We show that both, switches and lamps, increase the distinguishing power of the stochastic trace machine, respectively.

Trace equivalences are located at the bottom of the linear time - branching time spectrum for LTSs [vGla90]. In the setting of probabilistic automata, it has been shown by Segala that trace equivalence is coarser than relations based on testing or bisimulation [Seg96, Seg95]. However, if processes with nondeterminism act in continuous-time this is no longer valid. For instance, trace equivalence is incomparable to testing equivalence (compare Proposition 7.2). The reason for that is that schedulers can determine environment

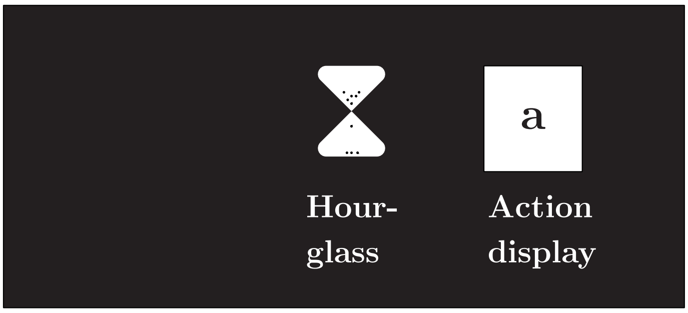

Figure 5.1: The stochastic trace machine

conditions which cannot be simulated by an environment in which external stimuli are exclusively provided by other processes. This fact is discussed more detailed in Chapter 6.

The trace equivalences defined in the sequel fail to be preserved by parallel composition of SPTPs, even in the purely interleaving case (without communication). This "non-property" carries over from the discrete-time setting [Seg95].

5.2 Trace Equivalence

Let us start with the description of the stochastic trace machine: Assume that phase type process $\mathcal{P}$ is represented by a machine which is essentially a black box. The box simulates the temporal evolution of $\mathcal{P}$ and displays $\mathcal{P}$'s visible behavior. As illustrated in Figure 5.1, it is equipped with two features:

- An *action display* shows the sequence of external actions performed by the process during a run of the machine.
- An *hourglass timer* counts down from a value initially specified by an external observer.

A run of the machine starts with the choice of an initial state with respect to the initial distribution ν of $\mathcal{P}$. The action display is empty at the beginning of the experiment. Then the machine behaves according to $\mathcal{P}$'s underlying transition system while the timer counts down. If a visible action is performed, it appears at the display. The action display remains unchanged until the next external action is performed by the process and shown at the action display. If a deadlock state is reached or the process diverges the action display still shows the symbol of the last visible action executed by $\mathcal{P}$ or remains empty if no visible action is carried out during the whole run. Note that the observer cannot distinguish between the case that $\mathcal{P}$ deadlocks or diverges. The observer records the sequence of displayed actions (where we assume that he can distinguish between two successive actions that are equal). The result is the trace of a path (compare Definition 3.4 on page 3.4). The experiment is over when the hourglass timer expires (i.e. the upper bulb of the hourglass is empty). Then the machine is reset for another run. This means, there is a countdown in each run and therefore a time interval can be associated with each trace (i.e. with each sequence of external actions recorded by the observer). Upon reset, the action display is cleared and the hourglass is turned (we assume that the observer works with different hourglasses to observe the process behavior for periods of time of different lengths). Then the machine starts again according to ν for another run and again the observer records the sequence of displayed actions until the hourglass timer expires etc.

If the machine encounters nondeterminism then it is resolved in the same way in each run, i.e. we fix a scheduler $\mathcal{D}$ for the whole experiment and restart the process infinitely often under $\mathcal{D}$. For each experiment we can deduce an observation function that gives the probability of each pair of trace and time bound (value of the hourglass timer).

In the sequel, we analyze trace semantics for different classes of schedulers and compare the respective equivalences with each other.

Let D be a class of schedulers, i.e.

$$D \in \{\mathsf{THR}, \mathsf{THD}, \mathsf{HR}, \mathsf{HD}, \mathsf{SR}, \mathsf{SD}, \mathsf{tTHR}, \ldots, \mathsf{tSD}\}.$$

And recall that $D(\mathcal{P})$ denotes the set of all D-schedulers for $\mathcal{P}$, in particular $\mathsf{THR}(\mathcal{P})$ denotes the set of all schedulers for $\mathcal{P}$. Now, consider PTP $\mathcal{P}$ under D-scheduler $\mathcal{D}$. Let $\sigma \in \mathsf{Act}^*$. We are interested in the measure of all paths $\pi \in \mathsf{path}(\mathcal{P})$ on which σ is performed within t time units.

Definition 5.1 (Trace Observation)
Let $\mathcal{P}$ be a PTP, D a scheduler class and $\mathcal{D} \in D(\mathcal{P})$. A *trace observation* is a function $\mathsf{tr}_{\mathcal{P}}^{\mathcal{D}} : (\mathsf{Act}^* \times \mathbb{R}_{\geq 0}) \to [0, 1]$ such that

$$\begin{aligned}\mathsf{tr}_{\mathcal{P}}^{\mathcal{D}}(\sigma, t) \quad = \quad & \mathsf{Pr}^{\mathcal{D}}(\{\pi \in \mathsf{path}(\mathcal{P}) \mid \pi \text{ is a } \mathcal{D}\text{-path and} \\ & \exists i : \mathit{trace}(\pi{\downarrow_i}) = \sigma, \mathit{time}(\pi{\downarrow_i}) \leq t\}).\end{aligned}$$

The *set of trace observations* with respect to scheduler class D is defined as

$$\mathsf{O}_{\mathsf{tr}}^{D}(\mathcal{P}) = \{\mathsf{tr}_{\mathcal{P}}^{\mathcal{D}} \mid \mathcal{D} \in D(\mathcal{P})\}.$$

Definition 5.2 (Trace Equivalence)
Two PTPs $\mathcal{P}_1$ and $\mathcal{P}_2$ are *trace equivalent* with respect to scheduler class D, written $\mathcal{P}_1 =_{\mathsf{tr}}^{D} \mathcal{P}_2$, if and only if

$$\mathsf{O}_{\mathsf{tr}}^{D}(\mathcal{P}_1) = \mathsf{O}_{\mathsf{tr}}^{D}(\mathcal{P}_2).$$

Remark 5.1
For trace semantics we consider $\mathcal{P}$'s behavior in different environments (i.e. simulated by different schedulers). We do not add information about them to the respective observations. Another process can "match" this behavior in an environment that is not necessarily the same as for $\mathcal{P}$. This means that two PTP*s $\mathcal{P}$ and $\mathcal{Q}$ are trace equivalent iff for each scheduler $\mathcal{D}$ for $\mathcal{P}$ there is a scheduler $\mathcal{D}'$ for $\mathcal{Q}$ that yields a matching observation and vice versa. This viewpoint is the core of trace semantics. Finer relations are obtained by equipping the stochastic trace machine with additional features such as action buttons that, if pressed by the observer, block (or weight) the occurrence of certain external actions. In this case, the recorded sequence consists of the performed trace* and *the sequence of (sets of) pushed action*

buttons (or the weights assigned to external actions). An equivalent process has to show the same behavior for the same sequence of blocked actions (compare Section 6.2 and 6.3).

Our next objective is to analyze the relationship between $=_{\mathsf{tr}}^{\mathsf{HR}}$ and the trace equivalence for probabilistic automata defined in [Seg95].

Definition 5.3 (Probabilistic Trace Equivalence)
Two PTPs $\mathcal{P}_1$ and $\mathcal{P}_2$ are *probabilistic trace equivalent*, written $\mathcal{P}_1 =_{\mathsf{ptr}}^{\mathsf{HR}} \mathcal{P}_2$, iff for each $\mathcal{D}_1 \in \mathsf{HR}(\mathcal{P}_1)$ there exists $\mathcal{D}_2 \in \mathsf{HR}(\mathcal{P}_2)$ such that

$$\lim_{t\to\infty} \mathsf{tr}_{\mathcal{P}_1}^{\mathcal{D}_1}(\sigma, t) = \lim_{t\to\infty} \mathsf{tr}_{\mathcal{P}_2}^{\mathcal{D}_2}(\sigma, t) \quad (\forall \sigma \in \mathsf{Act}^*).$$

Proposition 5.1
Let $\mathcal{P}_1$ and $\mathcal{P}_2$ be PTP*s.*

1. $\mathcal{P}_1 =_{\mathsf{tr}}^{\mathsf{HR}} \mathcal{P}_2$ *implies* $\mathcal{P}_1 =_{\mathsf{ptr}}^{\mathsf{HR}} \mathcal{P}_2$.

2. *If $\mathcal{P}_1$ and $\mathcal{P}_2$ are probabilistic automata then*

$$\mathcal{P}_1 =_{\mathsf{tr}}^{\mathsf{HR}} \mathcal{P}_2 \textit{ iff } \mathcal{P}_1 =_{\mathsf{ptr}}^{\mathsf{HR}} \mathcal{P}_2.$$

Let us consider a simple example which shows that the reverse of the first statement in the above proposition is not valid, i.e. probabilistic trace equivalence does not imply stochastic trace equivalence.

Example 5.1
Assume that PTP $\mathcal{P}$ consists of three states, say, s_1, s_2 and s_3 and transitions $s_1 \xrightarrow{\boldsymbol{\alpha}, T} s_2$, $s_2 \xrightarrow{a} s_3$ (recall that we simply write s instead of target distribution δ_s). Furthermore, initially $\mathcal{P}$ starts in s_1, i.e. $\nu(s_1) = 1$. Now, let $\mathcal{Q}$ be a copy of $\mathcal{P}$ except that representation $(\boldsymbol{\alpha}, T)$ is replaced by $(\boldsymbol{\beta}, V)$ with $F_{(\boldsymbol{\alpha},T)} \neq F_{(\boldsymbol{\beta},V)}$. Then $\mathcal{P} =_{\mathsf{ptr}}^{\mathsf{HR}} \mathcal{Q}$ but $\mathcal{P}$ and $\mathcal{Q}$ are not related by any of the stochastic trace equivalences $=_{\mathsf{tr}}^{D}$. Since $F_{(\boldsymbol{\alpha},T)} \neq F_{(\boldsymbol{\beta},V)}$ there exists

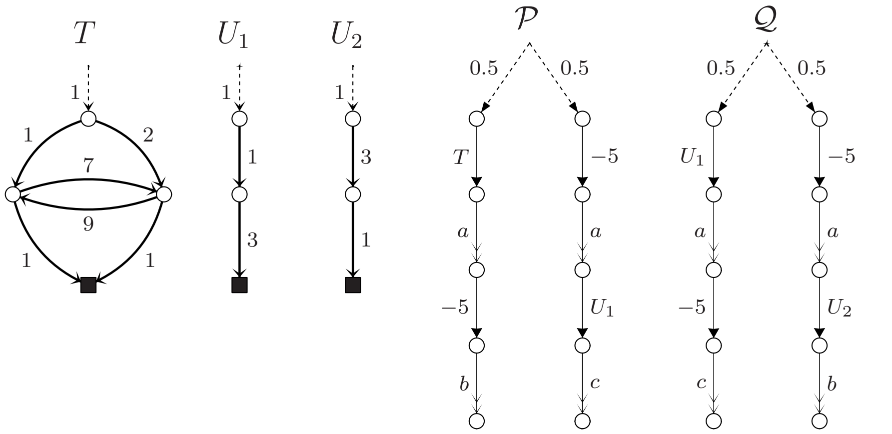

Figure 5.2: $F_T = F_{U_1} = F_{U_2}$ and $\mathcal{P} =^{D}_{\mathsf{tr}} \mathcal{Q}$ for all scheduler classes D.

$t > 0$ such that $F_{(\boldsymbol{\alpha},T)}(t) \neq F_{(\boldsymbol{\beta},V)}(t)$ which implies $\mathsf{tr}^{\mathcal{D}}_{\mathcal{P}}(a,t) \neq \mathsf{tr}^{\mathcal{D}'}_{\mathcal{Q}}(a,t)$ for all combinations of D-schedulers $\mathcal{D}$ and $\mathcal{D}'$.

Let us now examine a more complex example.

Example 5.2
Consider the two PTPs $\mathcal{P}$ and $\mathcal{Q}$ and the PH representations illustrated in Figure 5.2. For simplicity all target distributions are Dirac distributions and omitted in the illustration. Recall that in the case of illustrations of PH representations we depict the intensity graph in which the edges are labeled by the positive entries of the generator matrix whereas the single phased transitions of PTPs are labeled by negative parameters (because they constitute a special case of a PH representation). By computing the first three moments it can be shown that the phase type distributions of $\mathcal{P}$ and $\mathcal{Q}$ fulfill $F_T = F_{U_1} = F_{U_2}$ (compare Proposition 2.1 on page 27). Assume that scheduler $\mathcal{D} \in \mathsf{HD}(\mathcal{P})$ is such that in each state the single outgoing action transition is chosen with probability one. Similarly, $\mathcal{D}'$ chooses the available transitions of $\mathcal{Q}$ labeled by a, b and c with probability one. Then for both processes the probability of performing trace $\sigma = a$ within $t > 0$ time units

is given by

$$\begin{aligned}\mathsf{tr}_{\mathcal{P}}^{\mathcal{D}}(a,t) &= 0.5\cdot F_T(t) + 0.5\cdot F_{-5}(t)\\ &= 0.5\cdot F_{U_1}(t) + 0.5\cdot F_{-5}(t) = \mathsf{tr}_{\mathcal{Q}}^{\mathcal{D}'}(a,t).\end{aligned}$$

For trace $\sigma = ab$ we exploit the commutativity of the convolution of two random variables (compare Definition 2.5 on page 25) and get

$$\begin{aligned}\mathsf{tr}_{\mathcal{P}}^{\mathcal{D}}(ab,t) &= 0.5\cdot (F_T * F_{-5})(t)\\ &= 0.5\cdot (F_{-5} * F_{U_2})(t) = \mathsf{tr}_{\mathcal{Q}}^{\mathcal{D}'}(ab,t).\end{aligned}$$

Finally,

$$\begin{aligned}\mathsf{tr}_{\mathcal{P}}^{\mathcal{D}}(ac,t) &= 0.5\cdot (F_{-5} * F_{U_1})(t)\\ &= 0.5\cdot (F_{U_1} * F_{-5})(t) = \mathsf{tr}_{\mathcal{Q}}^{\mathcal{D}'}(ac,t).\end{aligned}$$

Now, if $\mathcal{D}$ decides to block the execution of b or c with a certain probability there is always a scheduler $\mathcal{D}' \in \mathsf{HD}(\mathcal{Q})$ that can "match" the observation $\mathsf{tr}_{\mathcal{P}}^{\mathcal{D}}$. Actually, it is not difficult to see that for all scheduler classes $\mathcal{P} =_{\mathsf{tr}}^{D} \mathcal{Q}$.

Let us now treat the relationship between $=_{\mathsf{bs}}$ and $=_{\mathsf{tr}}^{\mathsf{THR}}$.

Proposition 5.2

$=_{\mathsf{bs}}$ *is strictly finer than* $=_{\mathsf{tr}}^{\mathsf{THR}}$.

Proof. The inclusion $=_{\mathsf{bs}} \subset =_{\mathsf{tr}}^{\mathsf{THR}}$ follows directly from Theorem 3.1 (compare page 61) since the set of all paths on which trace σ is performed within a certain time interval is the (disjoint) union of sets $\Xi \in \mathcal{H}^R$ (where R is the phase type bisimulation relating $\mathcal{P}_1$ and $\mathcal{P}_2$).

An example that proves strictness is given in Figure 5.2. As already shown above we have $\mathcal{P} =_{\mathsf{tr}}^{\mathsf{THR}} \mathcal{Q}$. But $\mathcal{P} \neq_{\mathsf{bs}} \mathcal{Q}$ because there is no phase type bisimulation that relates the target states of the a-transitions. Thus, the initial states are not related and $\nu_{\mathcal{P}} \not\equiv_R \nu_{\mathcal{Q}}$ for all phase type bisimulations R. □

Note that $=_{\mathsf{bs}}$ is sensitive to τ-transitions whereas $=_{\mathsf{tr}}^{\mathsf{THR}}$ is not but our counterexample relies not on invisible transitions (otherwise, $=_{\mathsf{tr}}^{\mathsf{THR}} \not\subset =_{\mathsf{bs}}$ follows trivially).

Remark 5.2
A closer look at Example 5.2 shows that a finer notion of trace equivalence can be obtained if the observer records the time after each visible step, i.e. the amount of time that is needed to perform the next visible action is recorded. In this case, $\mathcal{P}$ and $\mathcal{Q}$ are distinguished. For trace $\sigma = ab$, for example, the probability to perform action a within, say, $t_1 = 0.25$ time units and afterwards action b within $t_2 = 1$ time units is much greater in the case of $\mathcal{Q}$ than in the case of $\mathcal{P}$. This can be seen by observing that

$$\underbrace{0.5 \cdot F_{-5}(0.25) \cdot F_{U_2}(1) \approx 0.169}_{\text{for } \mathcal{Q}} > \underbrace{0.034 \approx 0.5 \cdot F_T(0.25) \cdot F_{-5}(1)}_{\text{for } \mathcal{P}}.$$

In the sequel, we do not investigate this variant and restrict for simplicity to a single time bound.

5.3 Completed Trace Equivalence

A lot of results concerning trace semantics carry over from the nonprobabilistic setting to the probabilistic setting. An example is the observation that restricting to total schedulers increases the distinguishing power of the resulting equivalences. In the nonprobabilistic setting van Glabbeek defines *completed trace equivalence* by changing the features of the stochastic trace machine [vGla90]: Each time the process reaches a deadlock state, a special symbol is shown at the action display. This ensures the detection of states from which no visible action can be executed in the future. In our setting, we find such states if no further actions are observed from a certain time instant on, although the chosen scheduler is total. Recall that we only consider divergence free processes and therefore deadlock cannot be mixed up with divergence.

Remark 5.3
Completed trace equivalence, also called maximal trace equivalence, *coincides with trace equivalence in the fully probabilistic setting. For discrete-time models without nondeterminism this is formulated in [JS90] and [HT92b] and for the continuous-time, deterministic case compare [WMB05]. However, in the presence of nondeterminism this result is neither valid for the nonprobabilistic nor for the stochastic case.*

Proposition 5.3
Let $D \in \{\mathsf{THR}, \mathsf{THD}, \mathsf{HR}, \mathsf{HD}, \mathsf{SR}, \mathsf{SD}\}$ *Then*

$$=^{\mathsf{t}D}_{\mathsf{tr}} \text{ is strictly finer than } =^{D}_{\mathsf{tr}} .$$

Proof. We present a simple counterexample which proves $=^{\mathsf{HR}}_{\mathsf{tr}} \not\subseteq =^{\mathsf{tHR}}_{\mathsf{tr}}$ in CCS notation: The LTSs $a+(a.b)$ and $a.b.$ can be distinguished by completed trace equivalence (with respect to all scheduler classes of total schedulers). The observation of a single a can be made only in case of $a + (a.b)$ because in $a.b$ all total schedulers are forced to proceed with b. This example can also be found in [vGla90].
Now, we show that $\mathcal{P}_1 =^{\mathsf{t}D}_{\mathsf{tr}} \mathcal{P}_2$ implies $\mathcal{P}_1 =^{D}_{\mathsf{tr}} \mathcal{P}_2$ as follows: For $j \in \{1, 2\}$, we construct PTP $\mathcal{P}'_j$ from $\mathcal{P}_j$ by inserting a new state $stop_j$ and additional τ-transitions. These transitions have target distribution δ_{stop_j} and emanate from every state in which a total scheduler always assigns a distribution, i.e. all states s with $s \longrightarrow\!\!\!\!\rightarrow$ and $s \not\longrightarrow$. Then clearly, $\mathcal{P}_1 =^{\mathsf{t}D}_{\mathsf{tr}} \mathcal{P}_2$ implies $\mathcal{P}'_1 =^{\mathsf{t}D}_{\mathsf{tr}} \mathcal{P}'_2$. Assume $\mathcal{D}_1 \in D(\mathcal{P}_1)$. Now, a total scheduler $\mathcal{D}'_1 \in \mathsf{t}D(\mathcal{P}'_1)$ is constructed from $\mathcal{D}_1$ by choosing

$$\mathcal{D}'_1(\xi) = \begin{cases} \mathcal{D}_1(\xi) & \text{if } last(\xi) \not\longrightarrow\!\!\!\!\rightarrow \text{ or } last(\xi) \longrightarrow, \\ \lambda_\xi & \text{otherwise,} \end{cases}$$

where $\lambda_\xi(\tau, \delta_{stop_1}) = \mathcal{D}_1(\xi)^\perp$ and $\lambda_\xi(a, \mu) = \mathcal{D}_1(\xi)(a, \mu)$ for all a and all μ.

From $\mathcal{P}_1' =^{\mathsf{t}D}_{\mathsf{tr}} \mathcal{P}_2'$ we know that there is a total scheduler $\mathcal{D}_2'$ for $\mathcal{P}_2'$ that matches the observation associated with $\mathcal{D}_1'$. A scheduler $\mathcal{D}_2 \in D(\mathcal{P}_2)$ is obtained by transforming $\mathcal{D}_2'$ such that

$$\mathcal{D}_2(\xi) = \begin{cases} \mathcal{D}_2'(\xi) & \text{if } last(\xi) \not\twoheadrightarrow \text{ or } last(\xi) \longrightarrow, \\ \lambda_\xi & \text{otherwise,} \end{cases}$$

where $\lambda_\xi^\perp = \mathcal{D}_2'(\xi)(\tau, \delta_{stop_2})$ and $\lambda_\xi(a,\mu) = \mathcal{D}_2'(\xi)(a,\mu)$ for all a and all μ. Obviously, the respective observations of $\mathcal{D}_1$ and $\mathcal{D}_2$ are equal, i.e. $\mathsf{tr}_{\mathcal{P}_1}^{\mathcal{D}_1} = \mathsf{tr}_{\mathcal{P}_2}^{\mathcal{D}_2}$. In a similar way for each $\mathcal{D}_2 \in D(\mathcal{P}_2)$ we can construct a matching scheduler $\mathcal{D}_1 \in D(\mathcal{P}_1)$. We conclude that $\mathcal{P}_1 =^{D}_{\mathsf{tr}} \mathcal{P}_2$.

□

5.4 The Influence of Schedulers

The following theorem states the results of a comparison between the trace equivalence relations $=^{D}_{\mathsf{tr}}$. As we will see, many combinations $(=^{D}_{\mathsf{tr}}, =^{D'}_{\mathsf{tr}})$, $D \neq D'$ are incomparable. Nevertheless, the following observations can be made[2]

- HR is the scheduler class that leads to a relation being strictly coarser than most of the remaining trace equivalences:

$$=^{\mathsf{THR}}_{\mathsf{tr}}, =^{\mathsf{THD}}_{\mathsf{tr}} \text{ and } =^{\mathsf{HD}}_{\mathsf{tr}} \text{ are all strictly finer than } =^{\mathsf{HR}}_{\mathsf{tr}}$$

- If the scheduler decision additionally depends on timing information relations tend to become strictly finer:

$$=^{\mathsf{THD}}_{\mathsf{tr}} \subset =^{\mathsf{HD}}_{\mathsf{tr}} \text{ and } =^{\mathsf{THR}}_{\mathsf{tr}} \subset =^{\mathsf{HR}}_{\mathsf{tr}}$$

- Randomization decreases the distinguishing power (except in the stationary case):

$$=^{\mathsf{HD}}_{\mathsf{tr}} \subset =^{\mathsf{HR}}_{\mathsf{tr}} \text{ and } =^{\mathsf{THD}}_{\mathsf{tr}} \subset =^{\mathsf{THR}}_{\mathsf{tr}}$$

[2]The inclusions $=^{\mathsf{THD}}_{\mathsf{tr}} \subset =^{\mathsf{THR}}_{\mathsf{tr}}$ and $=^{\mathsf{HD}}_{\mathsf{tr}} \subset =^{\mathsf{HR}}_{\mathsf{tr}}$ are only shown for the case that the schedulers choose no visible action after a fixed number of steps (see Lemma 5.5).

The following theorem summarizes the influence scheduler classes have on trace equivalence.

Theorem 5.1 (Influence of Schedulers)

a) Time independent case:

Let $D, D' \in \{\mathsf{HR}, \mathsf{HD}, \mathsf{SR}, \mathsf{SD}\}$ *and* $D \neq D'$*. The relations* $=_{\mathsf{tr}}^{D}$ *and* $=_{\mathsf{tr}}^{D'}$ *are incomparable with the following two exceptions*[2]*:*

- $=_{\mathsf{tr}}^{\mathsf{HD}}$ *is strictly finer than* $=_{\mathsf{tr}}^{\mathsf{HR}}$*,*
- $=_{\mathsf{tr}}^{\mathsf{HR}} \not\subseteq =_{\mathsf{tr}}^{\mathsf{SR}}$ *but the opposite direction is unknown.*

b) Time dependent case:

- $=_{\mathsf{tr}}^{\mathsf{THD}}$ *is incomparable to* $=_{\mathsf{tr}}^{\mathsf{SR}}$ *and* $=_{\mathsf{tr}}^{\mathsf{SD}}$*,*
  but strictly finer than $=_{\mathsf{tr}}^{\mathsf{THR}}$*,* $=_{\mathsf{tr}}^{\mathsf{HD}}$ *and* $=_{\mathsf{tr}}^{\mathsf{HR}}$*,*
- $=_{\mathsf{tr}}^{\mathsf{THR}}$ *is incomparable to* $=_{\mathsf{tr}}^{\mathsf{SD}}$*,* $=_{\mathsf{tr}}^{\mathsf{SR}}$ *and* $=_{\mathsf{tr}}^{\mathsf{HD}}$*,*
  but strictly finer than $=_{\mathsf{tr}}^{\mathsf{HR}}$*.*

The remainder of this section focuses on the proofs of the results stated in the above theorem. Table 5.1 on page 95 gives an overview over the relationships

$$\text{“} =_{\mathsf{tr}}^{D_1} \overset{?}{\subseteq} =_{\mathsf{tr}}^{D_2} \text{”}$$

where each row corresponds to scheduler class D_1 and each column to class D_2. Entries that are pairs of PTPs refer to the counterexamples in Figure 5.3, 5.4 and 5.5 on page 96 to 99. Here, "$\subset$" denotes that D_1 is strictly finer than D_2. For some combinations more than one counterexample is indicated although only one is used in the proof.

The counterexamples showing the incomparableness in the case of time independent scheduler classes are mostly LTSs. Intuitively, this is due to the

$D_1 \setminus D_2$	THR	THD	HR	HD	SR	SD
THR	–	$\mathcal{P}_5, \mathcal{P}_6$	$\subset$	$\mathcal{P}_5, \mathcal{P}_6$	$\mathcal{P}_9, \mathcal{P}_{10}$	$\mathcal{P}_5, \mathcal{P}_6$
THD	$\subset$	–	$\subset$	$\subset$	$\mathcal{P}_1, \mathcal{P}_2$	$\mathcal{P}_1, \mathcal{P}_2$
HR	$\mathcal{P}_7, \mathcal{P}_8$	$\mathcal{P}_5, \mathcal{P}_6$ $\mathcal{P}_7, \mathcal{P}_8$	–	$\mathcal{P}_5, \mathcal{P}_6$	$\mathcal{P}_9, \mathcal{P}_{10}$	$\mathcal{P}_5, \mathcal{P}_6$
HD	$\mathcal{P}_7, \mathcal{P}_8$	$\mathcal{P}_7, \mathcal{P}_8$	$\subset$	–	$\mathcal{P}_1, \mathcal{P}_2$	$\mathcal{P}_1, \mathcal{P}_2$
SR	$\mathcal{P}_7, \mathcal{P}_8$	$\mathcal{P}_5, \mathcal{P}_6$ $\mathcal{P}_7, \mathcal{P}_8$	?	$\mathcal{P}_5, \mathcal{P}_6$	–	$\mathcal{P}_5, \mathcal{P}_6$
SD	$\mathcal{P}_3, \mathcal{P}_4$ $\mathcal{P}_7, \mathcal{P}_8$	$\mathcal{P}_3, \mathcal{P}_4$ $\mathcal{P}_7, \mathcal{P}_8$	$\mathcal{P}_3, \mathcal{P}_4$	$\mathcal{P}_3, \mathcal{P}_4$	$\mathcal{P}_3, \mathcal{P}_4$	–

Table 5.1: An overview of trace equivalence relationships[2]

fact that schedulers have no influence on the outcome of a race between PH transitions. The following lemma states the result of the last row and the last column of Table 5.1.

Lemma 5.1
$=^{\mathsf{SD}}_{\mathsf{tr}}$ *is incomparable to all other equivalences* $=^{D}_{\mathsf{tr}}, D \neq \mathsf{SD}$.

Proof. First, recall that nodes correspond to states, transitions of discrete distributions are illustrated as dashed edges, action and PH transitions as solid edges. The former type of transition is headed by two arrows, the latter by one arrow. Moreover, Dirac distributions are omitted.

Now, consider Figure 5.3. It is easy to see that $\mathcal{P}_3 =^{\mathsf{SD}}_{\mathsf{tr}} \mathcal{P}_4$ since for both

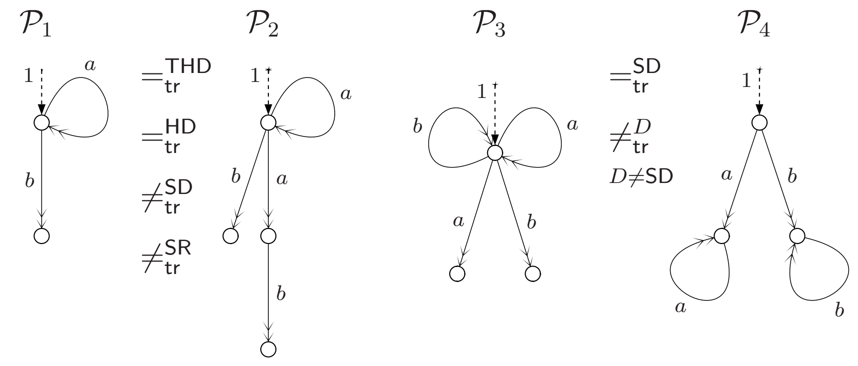

Figure 5.3: Counterexamples of Lemma 5.1 and 5.2 .

processes a SD-scheduler can select trace σ with positive probability if σ is of the form a^∞, b^∞, a or b. But then $\mathcal{P}_3 \neq_{\text{tr}}^{\text{SR}} \mathcal{P}_4$ since a scheduler $D \in \text{SR}(\mathcal{P}_3)$ can choose each loop with probability $\frac{1}{4}$ and the remaining two transitions also with probability $\frac{1}{4}$. Then traces of the form $(ab)^*$ have positive probability. This is not possible in the case of $\mathcal{P}_4$. Thus, $=_{\text{tr}}^{\text{SD}} \not\subseteq =_{\text{tr}}^{\text{SR}}$. With a similar argumentation it can be shown that $\mathcal{P}_3 \neq_{\text{tr}}^{\text{HR}} \mathcal{P}_4$ and $\mathcal{P}_3 \neq_{\text{tr}}^{\text{THR}} \mathcal{P}_4$ and therefore $=_{\text{tr}}^{\text{SD}} \not\subseteq =_{\text{tr}}^{\text{HR}}, =_{\text{tr}}^{\text{THR}}$.

We show $=_{\text{tr}}^{\text{SD}} \not\subseteq =_{\text{tr}}^{\text{HD}}, =_{\text{tr}}^{\text{THD}}$ by observing that $\mathcal{P}_3 \neq_{\text{tr}}^{\text{HD}} \mathcal{P}_4$ and $\mathcal{P}_3 \neq_{\text{tr}}^{\text{THD}} \mathcal{P}_4$. This can be seen by considering $D \in \text{HD}(\mathcal{P}_3)$ that produces sequences $\sigma \in (ab)^*$ with a positive probability by alternately choosing a and b. There is no HD-scheduler for which $\sigma \in (ab)^*$ has positive probability in $\mathcal{P}_4$. Since $\mathcal{P}_3$ and $\mathcal{P}_4$ are time-abstract, the same argumentation can be used for time dependent schedulers.

In what follows, we consider $=_{\text{tr}}^{\text{HD}}, =_{\text{tr}}^{\text{THD}} \not\subseteq =_{\text{tr}}^{\text{SD}}$. Let us again concentrate on Figure 5.3. First, note that $\mathcal{P}_1 =_{\text{tr}}^{\text{HD}} \mathcal{P}_2$ (and $\mathcal{P}_1 =_{\text{tr}}^{\text{THD}} \mathcal{P}_2$) since each possible branch of $\mathcal{P}_2$ can be matched by a scheduler for $\mathcal{P}_1$, e.g., if $\mathcal{P}_2$'s self-loop is never entered and the scheduler chooses first a and then b, a scheduler for $\mathcal{P}_1$ can choose the self-loop at first and the b-transitions next. But on the other hand $\mathcal{P}_1 \neq_{\mathsf{tr}}^{\mathsf{SD}} \mathcal{P}_2$ since trace $\sigma = ab$ can have a non-zero probability only in the case of $\mathcal{P}_2$.

Finally, we show $=_{\mathsf{tr}}^{\mathsf{SR}}, =_{\mathsf{tr}}^{\mathsf{HR}}, =_{\mathsf{tr}}^{\mathsf{THR}} \not\subseteq =_{\mathsf{tr}}^{\mathsf{SD}}$ as follows: In Figure 5.4 every SD-scheduler for $\mathcal{P}_6$ decides either for the left transition (label a) or for the right transition (label b) depending on the sojourn time of the initial state. But an element of $\mathsf{SD}(\mathcal{P}_5)$ can always choose the left a-transition in the left PH successor state and the right b-transition in the right PH successor state. The resulting observation cannot be matched by some $D \in \mathsf{SD}(\mathcal{P}_6)$. Therefore, $\mathcal{P}_5 \neq_{\mathsf{tr}}^{\mathsf{SD}} \mathcal{P}_6$.

We have $(\mathcal{P}_5, \mathcal{P}_6) \in =_{\mathsf{tr}}^{\mathsf{THR}}$ because for a given residence time in the initial state a scheduler for $\mathcal{P}_6$ can match every choice $\{p_l^a, p_l^b, p_r^a, p_r^b\}$ of $D \in \mathsf{THR}(\mathcal{P}_5)$ (where p_l^a is the probability to take the leftmost a-transition, etc.) by choosing the a-transition with probability $\frac{1}{2}p_l^a + \frac{1}{2}p_r^a$ and the b-transition with probability $\frac{1}{2}p_l^b + \frac{1}{2}p_r^b$.

The time-abstract cases can be shown using a similar argumentation, i.e. $\mathcal{P}_5 =_{\mathsf{tr}}^{\mathsf{HR}} \mathcal{P}_6$, $\mathcal{P}_5 =_{\mathsf{tr}}^{\mathsf{SR}} \mathcal{P}_6$. Thus, $=_{\mathsf{tr}}^{\mathsf{SR}}, =_{\mathsf{tr}}^{\mathsf{HR}}, =_{\mathsf{tr}}^{\mathsf{THR}} \not\subseteq =_{\mathsf{tr}}^{\mathsf{SD}}$.

□

Lemma 5.2

$=_{\mathsf{tr}}^{\mathsf{SR}}$ *is incomparable to all other equivalences* $=_{\mathsf{tr}}^{D}, D \neq \mathsf{SR}$ *except in the case of* $D = \mathsf{HR}$ *where* $=_{\mathsf{tr}}^{\mathsf{HR}} \not\subseteq =_{\mathsf{tr}}^{\mathsf{SR}}$*, but the other direction is unknown (compare the row and the column before last of Table 5.1).*

Proof. First recall from Lemma 5.1 that we already have shown that $=_{\mathsf{tr}}^{\mathsf{SD}}$ and $=_{\mathsf{tr}}^{\mathsf{SR}}$ are incomparable. For $=_{\mathsf{tr}}^{\mathsf{SR}} \not\subseteq =_{\mathsf{tr}}^{\mathsf{HD}}, =_{\mathsf{tr}}^{\mathsf{THD}}$ consider again the pair $(\mathcal{P}_5, \mathcal{P}_6)$ in Figure 5.4. As shown above $\mathcal{P}_5 =_{\mathsf{tr}}^{\mathsf{SR}} \mathcal{P}_6$. On the other hand $\mathcal{P}_5 \neq_{\mathsf{tr}}^{D} \mathcal{P}_6$ for $D \in \{\mathsf{HD}, \mathsf{THD}\}$ with a similar argumentation as in the proof of Lemma 5.1 where we showed that $\mathcal{P}_5 \neq_{\mathsf{tr}}^{\mathsf{SD}} \mathcal{P}_6$.

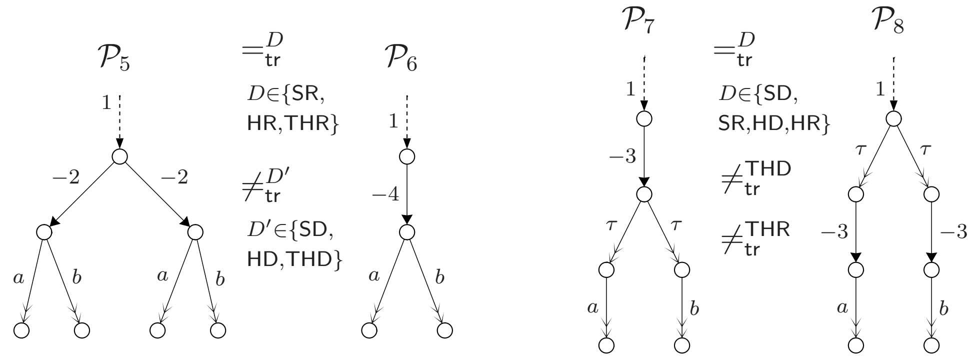

Figure 5.4: Counterexamples of Lemma 5.1, 5.2 and 5.3.

Let us focus on $=_{\mathsf{tr}}^{\mathsf{HD}}, =_{\mathsf{tr}}^{\mathsf{THD}} \not\subseteq =_{\mathsf{tr}}^{\mathsf{SR}}$. In Figure 5.3 we have $\mathcal{P}_1 =_{\mathsf{tr}}^{\mathsf{HD}} \mathcal{P}_2$ and $\mathcal{P}_1 =_{\mathsf{tr}}^{\mathsf{THD}} \mathcal{P}_2$ (compare again the proof of Lemma 5.1). But $\mathcal{P}_1 \neq_{\mathsf{tr}}^{\mathsf{SR}} \mathcal{P}_2$ because if for $\mathcal{P}_2$ a SR-scheduler chooses the a-transition to the lower state with a positive probability, there is no scheduler for $\mathcal{P}_1$ that can match the corresponding observation.

Finally, we show $=_{\mathsf{tr}}^{\mathsf{HR}}, =_{\mathsf{tr}}^{\mathsf{THR}} \not\subseteq =_{\mathsf{tr}}^{\mathsf{SR}}$ by considering $\mathcal{P}_9$ and $\mathcal{P}_{10}$ in Figure 5.5 on page 99. It holds that $\mathcal{P}_9 =_{\mathsf{tr}}^{\mathsf{HR}} \mathcal{P}_{10}$ (and also $\mathcal{P}_9 =_{\mathsf{tr}}^{\mathsf{THR}} \mathcal{P}_{10}$) because each choice for a or b of an HR-scheduler for $\mathcal{P}_9$ can be matched by a choice of an HR-scheduler for $\mathcal{P}_{10}$ (observe that in each state there is exactly one a-transition and one b-transition). But a SR-scheduler for $\mathcal{P}_9$ has only one choice whereas for $\mathcal{P}_{10}$ a SR-scheduler can choose three times, possibly always different subdistributions. This yields observations that cannot be matched by a SR-scheduler for $\mathcal{P}_9$ and thus $\mathcal{P}_9 \neq_{\mathsf{tr}}^{\mathsf{SR}} \mathcal{P}_{10}$. □

Remark 5.4

To the best of our knowledge, it is still an open problem if $=_{\mathsf{tr}}^{\mathsf{SR}} \subset =_{\mathsf{tr}}^{\mathsf{HR}}$ holds. For many optimization problems related to discrete time Markov decision processes *it is known that there exists an optimal scheduler being history independent (compare [Der70], for example). However, in the setting of trace observations, we remark that it is non-trivial to describe an* HR*-scheduler by*

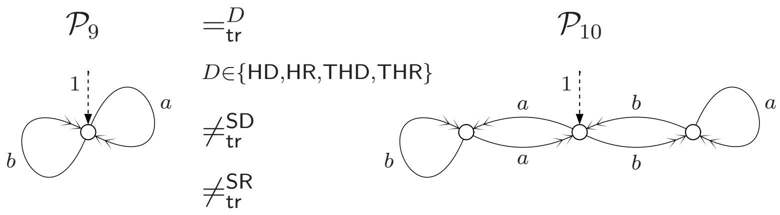

Figure 5.5: Counterexamples of Lemma 5.2 and 5.3.

combinations of SR*-schedulers such that* $=^{\mathsf{SR}}_{\mathsf{tr}} \subset =^{\mathsf{HR}}_{\mathsf{tr}}$ *can be established.*

Lemma 5.3

Both, $=^{\mathsf{HD}}_{\mathsf{tr}}$ *and* $=^{\mathsf{HR}}_{\mathsf{tr}}$ *are not contained in* $=^{\mathsf{THR}}_{\mathsf{tr}}$ *or* $=^{\mathsf{THR}}_{\mathsf{tr}}$*.*

Proof. Consider the two PTPs $\mathcal{P}_7$ and $\mathcal{P}_8$ in Figure 5.4 on page 98. In both cases, the observer will see action a after a delay that follows an exponential distribution with parameter -3 if the left branch is chosen, respectively. Trace b has the same distribution provided that the right branch is chosen. It is easy to see that $\mathcal{P}_7 =^{D}_{\mathsf{tr}} \mathcal{P}_8$ for all time-abstract scheduler classes D because a scheduler $\mathcal{D}' \in D(\mathcal{P}_8)$ can "match" the observations under $\mathcal{D} \in D(\mathcal{P}_7)$ by choosing the same distribution for the nondeterministic τ-transitions as $\mathcal{D}$. On the other hand $\mathcal{P}_7 \neq^{\mathsf{THD}}_{\mathsf{tr}} \mathcal{P}_8$ and $\mathcal{P}_7 \neq^{\mathsf{THR}}_{\mathsf{tr}} \mathcal{P}_8$, because $\mathcal{D} \in \mathsf{THD}(\mathcal{P}_7)$ can choose the left branch if the first delay (in the initial state) is lower than, say, $x = 1$ and the right branch if the delay is greater or equal 1. Then the probability $\mathsf{tr}^{\mathcal{D}}_{\mathcal{P}_7}(a, t)$ of trace a within t time units is zero if $t > 1$ but greater zero otherwise. There is no scheduler $\mathcal{D}' \in \mathsf{THR}(\mathcal{P}_8)$ that can match the observation $\mathsf{tr}^{\mathcal{D}}_{\mathcal{P}_7}$ since the nondeterministic branching in $\mathcal{P}_8$ occurs at a point where no time has passed yet. □

We need some preliminary observations to prove the next lemma.

Let $\mathcal{D} \in \mathsf{HR}(\mathcal{P})$ for some PTP $\mathcal{P} = (S_\mathcal{P}, \longrightarrow_\mathcal{P}, \twoheadrightarrow_\mathcal{P}, \nu_\mathcal{P})$, let $\sigma \in \mathsf{Act}^+$ and $f : \mathbb{R}_{\geq 0} \to [0,1]$ with

$$f(t) := \mathsf{tr}_\mathcal{P}^\mathcal{D}(\sigma, t).$$

We define $X_\sigma : \Omega \to \mathbb{R}_{\geq 0} \cup \{\infty\}$ as the random variable which describes the time that process $\mathcal{P}$ needs to perform trace σ (under $\mathcal{D}$), i.e.

$$Prob\big(X_\sigma \leq t\big) = f(t),$$

$$Prob\big(X_\sigma = \infty\big) = 1 - \lim_{t\to\infty} f(t).$$

Proposition 5.4

There exists a (possibly infinite) CTMC $(\boldsymbol{\nu}, Q)$ *such that the distribution of* X_σ *equals the distribution of the time to reach a certain set of states in* $(\boldsymbol{\nu}, Q)$.

Proof. We construct $(\boldsymbol{\nu}, Q)$ in two steps:

i) The first step is concerned with the definition of an HR-scheduler $\mathcal{D}_\sigma$ which decides like $\mathcal{D}$ but stops scheduling if either σ has already been performed or if σ cannot be performed in the future. Assume that $\sigma = \sigma' a$, i.e. $a \in \mathsf{Act}$ is the last element of σ (recall that $\sigma \neq \epsilon$). Let $\mathsf{pathf}(\sigma)$ denote the set of all prefixes of $\mathcal{D}$-path fragments $\xi = \xi' a s \in \mathsf{pathf}(\mathcal{P})$ with $trace(\xi') = \sigma'$, $s \in S$. We construct $\mathcal{D}_\sigma$ from $\mathcal{D}$ as follows:

$$\mathcal{D}_\sigma(\xi) := \begin{cases} \mathcal{D}(\xi) & \text{if } trace(\xi) \neq \sigma \text{ and } \xi \in \mathsf{pathf}(\sigma), \\ \lambda & \text{otherwise,} \end{cases}$$

where λ is such that $\lambda^\perp = 1$.

Let Y_σ be the random variable which describes the time until $\mathcal{P}$ performs trace σ under scheduler $\mathcal{D}_\sigma$. It is easy to verify that Y_σ and X_σ have the same distribution.

ii) We define a PTP $\mathcal{P}' = (S', \longrightarrow', \twoheadrightarrow', \nu')$ which is the (deterministic) tree describing the behavior of $\mathcal{P}$ under $\mathcal{D}_\sigma$. This means that each state of $\mathcal{P}'$ corresponds to an untimed history $\kappa = untime(\xi)$ where $\xi \in \mathsf{pathf}(\mathcal{P})$. In $\mathcal{P}'$ immediate transitions are collapsed, i.e. a path fragment which exclusively contains events $\mathsf{e} \in \mathsf{Act}_\tau$ is removed and integrated in the preceding timed step (or in the initial distribution if there is no such preceding step). The information about the actions taken is stored in the state's name (the history κ).

The set of states is given by $S' := S^{\mathcal{D}_\sigma} \cup \{s^\infty\}$ where

$$S^{\mathcal{D}_\sigma} := \{\kappa = untime(\xi) \mid \xi \in \mathsf{pathf}(\sigma) \wedge \big(\mathcal{D}_\sigma(\kappa)\big)^\perp > 0\}.$$

(Recall that we may take an untimed path fragment as argument of a time independent scheduler.) Let $\mathsf{inst}^{\mathcal{D}_\sigma} \subseteq \mathsf{pathf}(\mathcal{P})$ be the subset of untimed $\mathcal{D}_\sigma$-path fragments κ which encode an instantaneous path fragment ξ, i.e. $\kappa = untime(\xi) = \xi$. This means that no PH transitions have been taken on ξ. The initial distribution ν' is such that if $\kappa \in \mathsf{inst}^{\mathcal{D}_\sigma}$ we set

$$\nu'(\kappa) := \nu_{\mathcal{P}}(first(\kappa)) \cdot col(\kappa) \cdot \mathcal{D}_\sigma(\kappa)^\perp$$

where function $col : \mathsf{inst}^{\mathcal{D}_\sigma} \to [0,1]$ (col from "collapse") is inductively given by

$$\begin{aligned} col(s) &:= 1 \\ col(\kappa' a s) &:= col(\kappa') \cdot \sum_{\substack{\mu:\mu(s)>0,\\ last(\kappa') \xrightarrow{a} \mu}} \mathcal{D}_\sigma(\kappa')(a,\mu) \cdot \mu(s) \end{aligned}$$

for all $a \in \mathsf{Act}_\tau$, $s \in S_{\mathcal{P}}$. We put $\nu'(\kappa) := 0$ if $\kappa \in S^{\mathcal{D}_\sigma} \setminus \mathsf{inst}^{\mathcal{D}_\sigma}$ and let

$$\nu'(s^\infty) := 1 - \sum_{\kappa \in S^{\mathcal{D}_\sigma}} \nu'(\kappa).$$

The PH transitions of $\mathcal{P}'$ are defined as follows: $s^\infty \not\longrightarrow$ and also all states κ with $trace(\kappa) = \sigma$. For all remaining $\kappa \in S'$ we have that

$\kappa \xrightarrow{\boldsymbol{\alpha},T} \mu$ iff $last(\kappa) \xrightarrow{\boldsymbol{\alpha},T} \mu'$ where μ is such that for all $\kappa\hat{\kappa} \in S^{\mathcal{D}_\sigma}$ with $\hat{\kappa} \in \mathsf{inst}^{\mathcal{D}_\sigma}$ we have that[3]

$$\mu(\kappa\hat{\kappa}) = \mu'(first(\hat{\kappa})) \cdot col(\hat{\kappa}) \cdot \mathcal{D}_\sigma(\kappa\hat{\kappa})^{\perp}$$

and $\mu(s^\infty) = 1 - \sum_{\kappa' \in S^{\mathcal{D}}_\sigma} \mu(\kappa')$. Thus, state s^∞ represents all $\mathcal{D}$-path fragments of $\mathcal{P}$ which cannot be continued to a $\mathcal{D}$-path fragment with trace σ.

Then CTMC $(\boldsymbol{\nu}, Q)$ is given by $\boldsymbol{\nu} = \boldsymbol{\nu}_{S'}$ and Q is the generator $Q_{S'}$ of $\mathcal{P}'$ (compare Definition 3.3 on page 38). From construction, the time until an absorbing state κ with $trace(\kappa) = \sigma$ is reached has the same distribution as Y_σ and thus as X_σ. □

Lemma 5.4
$\mathcal{P} =^{\mathsf{THD}}_{\mathsf{tr}} \mathcal{Q}$ implies $\mathcal{P} =^{\mathsf{HD}}_{\mathsf{tr}} \mathcal{Q}$, and $\mathcal{P} =^{\mathsf{THR}}_{\mathsf{tr}} \mathcal{Q}$ implies $\mathcal{P} =^{\mathsf{HR}}_{\mathsf{tr}} \mathcal{Q}$.

Proof. We give the proof details only for the randomized case. The deterministic case can be shown by using a very similar argumentation.
We assume that $\mathcal{P} =^{\mathsf{THR}}_{\mathsf{tr}} \mathcal{Q}$ and consider a scheduler $\mathcal{D} \in \mathsf{HR}(\mathcal{P})$. Since $\mathsf{HR}(\mathcal{P}) \subset \mathsf{THR}(\mathcal{P})$ there exists $\mathcal{D}' \in \mathsf{THR}(\mathcal{Q})$ with $\mathsf{tr}^{\mathcal{D}}_{\mathcal{P}} = \mathsf{tr}^{\mathcal{D}'}_{\mathcal{Q}}$. We define scheduler $\mathcal{D}'' \in \mathsf{HR}(\mathcal{Q})$ as follows: Let $\pi \in \mathsf{path}(\mathcal{Q})$ with

$$\pi{\downarrow_i} = s_1 \xrightarrow{\mathsf{e}_1} s_2 \xrightarrow{\mathsf{e}_2} \ldots \xrightarrow{\mathsf{e}_{i-1}} s_i.$$

For $i > 1$ we define $\mathcal{D}''$'s decision by

$$\mathcal{D}''(s_1 \xrightarrow{\mathsf{e}_1} s_2 \xrightarrow{\mathsf{e}_2} \ldots \xrightarrow{\mathsf{e}_{i-1}} s_i) := \mathcal{D}'(s_1 \xrightarrow{\hat{\mathsf{e}}_1} s_2 \xrightarrow{\hat{\mathsf{e}}_2} \ldots \xrightarrow{\hat{\mathsf{e}}_{i-1}} s_i)$$

[3]Recall that in $untime(\xi)$ timed events are simply left out, i.e. we may have several successive states which are not separated by an action as e.g. in $untime(\xi) = \kappa\hat{\kappa} = s_1 a s_2 s_3 b s_4$, $\kappa = s_1 a s_2$ and $\hat{\kappa} = s_3 b s_4$.

where for $j \in \{1, 2, \ldots, i\}$

$$\hat{\mathsf{e}}_j = \begin{cases} \mathsf{e}_j & \text{if } \mathsf{e}_j \in \mathsf{Act}_\tau \\ 1 & \text{if } \mathsf{e}_j \in \mathbb{R}_{>0}. \end{cases}$$

We remark that instead of $\hat{\mathsf{e}}_j = 1$ we can choose an arbitrary but fixed event in $\mathbb{R}_{>0}$. If $i = 1$ we put $\mathcal{D}''(s_1) = \mathcal{D}'(s_1)$. Next we show that $\mathsf{tr}_{\mathcal{Q}}^{\mathcal{D}'} = \mathsf{tr}_{\mathcal{Q}}^{\mathcal{D}''}$. From the definition of schedulers we know that for each decision $\mathcal{D}'(\xi)$ there exists a non-empty interval of $\mathbb{R}_{\geq 0}$ on which the decisions of $\mathcal{D}'$ are constant. In combination with the definition of $\mathcal{D}''$ we conclude that for each $\sigma \in \mathsf{Act}^*$ there exists a non-empty interval $J_\sigma \subseteq \mathbb{R}_{\geq 0}$ such that

$$\mathsf{tr}_{\mathcal{Q}}^{\mathcal{D}'}(\sigma, t) = \mathsf{tr}_{\mathcal{Q}}^{\mathcal{D}''}(\sigma, t) \quad (\forall t \in J_\sigma).$$

Let us fix σ and for $k \in \{1, 2\}$ define $f_k : \mathbb{R}_{\geq 0} \to [0, 1]$ by

$$f_1(x) := \mathsf{tr}_{\mathcal{P}}^{\mathcal{D}}(\sigma, x) \text{ and } f_2(x) := \mathsf{tr}_{\mathcal{Q}}^{\mathcal{D}''}(\sigma, x)$$

for all $x \in \mathbb{R}_{\geq 0}$. We claim that

$$f_1(t) = f_2(t)\ (\forall t \in J_\sigma) \implies f_1(t) = f_2(t)\ (\forall t \in \mathbb{R}_{>0}). \tag{5.1}$$

From Proposition 5.4 we know that f_i has a Markov chain representation. Thus, f_i can be expressed in terms of transient state probabilities. However, this implies that f_i is an analytic function. But then from the identity theorem, the implication 5.1 is true, i.e. if f_1 and f_2 coincide on an arbitrary small interval they coincide everywhere on $\mathbb{R}_{\geq 0}$. □

For scheduler class D let D^k be the set of all D-schedulers that do not decide for visible actions after a path fragment of length k, i.e.

$$\mathcal{D} \in D^k \implies \forall a \in \mathsf{Act}, \forall \mu, \forall \xi, |\xi| \geq k : \mathcal{D}(\xi)(a, \mu) = 0.$$

Let $=_{\mathsf{tr}}^{D^k}$ be the induced trace equivalence of class D^k.

Lemma 5.5

Let $k \geq 1$. Then

$$=_{\mathsf{tr}}^{\mathsf{HD}^k} \subseteq =_{\mathsf{tr}}^{\mathsf{HR}^k} \quad \textit{and} \quad =_{\mathsf{tr}}^{\mathsf{THD}^k} \subseteq =_{\mathsf{tr}}^{\mathsf{THR}^k}$$

Proof. We show the statement for the time-abstract case and omit the timed case as the argumentation follows similar lines.

Let us first fix some notation. For an HR-scheduler $\mathcal{D} \in \mathsf{HR}(\mathcal{P})$, let $\mathcal{D}|_n$ denote the restriction of $\mathcal{D}$ on path fragments of length $n \in \mathbb{N}$ and let $\mathsf{HR}_n(\mathcal{P})$ be the set of all such partial schedulers $\mathcal{D}{\downarrow_n}$ for $\mathcal{P}$. Similarly, $\mathsf{HD}_n(\mathcal{P})$ denotes the subset of all restrictions of HD-schedulers for $\mathcal{P}$. Note that $\mathsf{HD}_n(\mathcal{P})$ is a finite set because $\mathcal{P}$ is finitely branching. Let $h := |\mathsf{HD}_n(\mathcal{P})|$ and let $\mathcal{D}$ be an HR scheduler for $\mathcal{P}$. The proof is separated into two steps:

1. We define the set $\Lambda_n(\mathcal{D})$ as follows:

$$\begin{aligned}\Lambda_n(\mathcal{D}) \quad := \quad & \Big\{ \big(x_{\mathcal{E}} \big)_{\mathcal{E} \in \mathsf{HD}_n(\mathcal{P})} \in [0,1]^h : \textstyle\sum_{\mathcal{E} \in \mathsf{HD}_n(\mathcal{P})} x_{\mathcal{E}} = 1 \\ & \text{and } \mathcal{D}|_n = \textstyle\sum_{\mathcal{E} \in \mathsf{HD}_n(\mathcal{P})} x_{\mathcal{E}} \cdot \mathcal{E} \Big\}\end{aligned} \tag{5.2}$$

   where $\mathcal{D}|_n$ and $\mathcal{E}$ are taken as (finite) vectors in which each entry corresponds to a path fragment of length $k \leq n$. The value $x_{\mathcal{E}}$ is the weight of the partial HD-scheduler $\mathcal{E}$ and scheduler $\mathcal{D}$ is "approximated" until its n-th decision. Note that $\Lambda_n(\mathcal{D})$ is a non-empty and compact set. Next we define for $m \geq n$ functions $f_{m,n} : \Lambda_m(\mathcal{D}) \to \Lambda_n(\mathcal{D})$ by

$$f_{m,n}\big((x_{\mathcal{E}})_{\mathcal{E} \in \mathsf{HD}_m(\mathcal{P})} \big) := (y_{\mathcal{F}})_{\mathcal{F} \in \mathsf{HD}_n(\mathcal{P})} \text{ where } y_{\mathcal{F}} := \sum_{\substack{\mathcal{E} \in \mathsf{HD}_m(\mathcal{P}), \\ \mathcal{F} = \mathcal{E}|_n}} x_{\mathcal{E}}$$

   to sum up certain entries of vector $(x_{\mathcal{E}})_{\mathcal{E} \in \mathsf{HD}_m(\mathcal{P})}$ and calculate weights for $\mathcal{F} \in \mathsf{HD}_n(\mathcal{P})$. Let

$$f_{m,n}\big(\Lambda_m(\mathcal{D}) \big) = \Big\{ f_{m,n}\big((x_{\mathcal{E}})_{\mathcal{E} \in \mathsf{HD}_m(\mathcal{P})} \big) \mid (x_{\mathcal{E}})_{\mathcal{E} \in \mathsf{HD}_m(\mathcal{P})} \in \Lambda_m(\mathcal{D}) \Big\}.$$

Obviously, each element in $f_{m,n}\big(\Lambda_m(\mathcal{D})\big)$ is a vector of size h. Our next objective is a backward construction for which we take into account all future decisions of $\mathcal{D}$. The set

$$\Gamma_n(\mathcal{D}) := \bigcap_{m:=n}^{\infty} f_{m,n}\big(\Lambda_m(\mathcal{D})\big)$$

is non-empty and compact because it is the intersection of non-empty compact sets and because $f_{m,n}$ is continuous. Our aim is now to pick out elements $z^{(0)}, z^{(1)}, \ldots$ of the sets $\Gamma_0(\mathcal{D}), \Gamma_1(\mathcal{D}), \ldots$ such that they constitute a sequence which converges uniformly. We choose $z^{(0)} \in \Gamma_0(\mathcal{D})$ arbitrary and let $z^{(n+1)} \in \Gamma_{n+1}(\mathcal{D})$ be such that $f_{n+1,n}(z^{(n+1)}) = z^{(n)}$. From construction we get for $\mathcal{E} \in \mathsf{HD}_{n+1}(\mathcal{P})$

$$z^{(n)}_{\mathcal{E}|_n} = \sum_{\substack{\mathcal{F} \in \mathsf{HD}_{n+1}(\mathcal{P}) \\ \mathcal{F}|_n = \mathcal{E}|_n}} z^{(n+1)}_{\mathcal{F}} \geq z^{(n+1)}_{\mathcal{E}} \tag{5.3}$$

where $z^{(m)} = \big(z^{(m)}_{\mathcal{E}}\big)_{\mathcal{E} \in \mathsf{HD}_m(\mathcal{P})}$ for $m \geq 1$.

It is clear that for each n we can find $z^{(n)} = \big(z^{(n)}_{\mathcal{E}}\big)_{\mathcal{E} \in \mathsf{HD}_n(\mathcal{P})}$ such that Equation 5.3 holds and additionally

$$\sum_{\mathcal{E} \in \mathsf{HD}_n(\mathcal{P})} z^{(n)}_{\mathcal{E}} = 1,\ 0 \leq z^{(n)}_{\mathcal{E}} \leq 1.$$

We conclude the first step by observing that for each path fragment $\xi \in \mathsf{pathf}(\mathcal{P})$ of length $m \leq n$ we can show by induction on m and n that

$$\mathsf{Pr}^{\mathcal{D}}(\xi) = \sum_{\mathcal{E} \in \mathsf{HD}_n(\mathcal{P})} z^{(n)}_{\mathcal{E}} \cdot \mathsf{Pr}^{\mathcal{E}}(\xi) \tag{5.4}$$

where the probability $\mathsf{Pr}^{\mathcal{E}}(\{\pi \mid \xi \text{ is a prefix of } \pi\}) = \mathsf{Pr}^{\mathcal{E}}(\xi)$ for partial HD-scheduler $\mathcal{E}$ is well-defined if $|\xi| \leq n$.

2. Let $f : \mathsf{HD}_k(\mathcal{P}) \to \mathsf{HD}_k(\mathcal{Q})$ be a function such that for all $\sigma \in \mathsf{Act}^*$, $t \geq 0$

$$\sum_{\substack{\xi:|\xi|\leq k\\ trace(\xi)=\sigma, time(\xi)\leq t}} \mathsf{Pr}^{\mathcal{E}}_{\mathcal{P}}(\xi) = \sum_{\substack{\xi:|\xi|\leq k\\ trace(\xi)=\sigma, time(\xi)\leq t}} \mathsf{Pr}^{f(\mathcal{E})}_{\mathcal{Q}}(\xi).$$

The existence of f follows from $\mathcal{P} =^{\mathsf{HD}^k}_{\mathsf{tr}} \mathcal{Q}$. Furthermore, for $\mathcal{E}' \in \mathsf{HD}_k(\mathcal{Q})$ we define

$$z^{(k)}_{\mathcal{E}'} := \sum_{\substack{\mathcal{E}\in\mathsf{HD}_k(\mathcal{P})\\ f(\mathcal{E})=\mathcal{E}'}} z^{(k)}_{\mathcal{E}}. \tag{5.5}$$

This yields for $\mathcal{D} \in \mathsf{HR}^k$

$$\begin{aligned}
\mathsf{tr}^{\mathcal{D}}_{\mathcal{P}}(\sigma,t) &= \sum_{\substack{\xi:|\xi|\leq k\\ trace(\xi)=\sigma, time(\xi)\leq t}} \mathsf{Pr}^{\mathcal{D}}(\xi)\\
&\overset{5.4}{=} \sum_{\substack{\xi:|\xi|\leq k\\ trace(\xi)=\sigma, time(\xi)\leq t}} \sum_{\mathcal{E}\in\mathsf{HD}_k(\mathcal{P})} z^{(k)}_{\mathcal{E}} \cdot \mathsf{Pr}^{\mathcal{E}}(\xi)\\
&= \sum_{\mathcal{E}\in\mathsf{HD}_k(\mathcal{P})} z^{(k)}_{\mathcal{E}} \sum_{\substack{\xi:|\xi|\leq k\\ trace(\xi)=\sigma, time(\xi)\leq t}} \mathsf{Pr}^{\mathcal{E}}(\xi)\\
&\overset{5.5}{=} \sum_{\mathcal{E}'\in\mathsf{HD}_k(\mathcal{Q})} z^{(k)}_{\mathcal{E}'} \sum_{\substack{\xi:|\xi|\leq k\\ trace(\xi)=\sigma, time(\xi)\leq t}} \mathsf{Pr}^{\mathcal{E}'}(\xi).
\end{aligned}$$

We define the partial HR-scheduler $\tilde{\mathcal{D}} \in \mathsf{HR}_k(\mathcal{Q})$ by

$$\tilde{\mathcal{D}} = \sum_{\mathcal{E}'\in\mathsf{HD}_k(\mathcal{Q})} z^{(k)}_{\mathcal{E}'} \cdot \mathcal{E}'$$

and get for all $\mathcal{D}' \in \mathsf{HR}^k$ with $\mathcal{D}'\downarrow_k = \tilde{\mathcal{D}}$ that

$$\mathsf{tr}^{\mathcal{D}}_{\mathcal{P}}(\sigma,t) = \sum_{\substack{\xi:|\xi|\leq k\\ trace(\xi)=\sigma, time(\xi)\leq t}} \mathsf{Pr}^{\tilde{\mathcal{D}}}(\xi) = \mathsf{tr}^{\mathcal{D}'}_{\mathcal{Q}}(\sigma,t)$$

for all $\sigma \in \mathsf{Act}^*$, $t \geq 0$. But then $\mathcal{P} =^{\mathsf{HR}^k}_{\mathsf{tr}} \mathcal{Q}$.

□

Unfortunately, the general case is much more difficult and there is no proof yet. The problem is that if there is no bound on the number of transitions until a certain trace is performed the second step of the proof can not be done as above. Nevertheless, we claim that the statement is true in the case of $=_{\mathsf{tr}}^{\mathsf{HR}}$ and $=_{\mathsf{tr}}^{\mathsf{THR}}$ as well.

All remaining relationships of Table 5.1 follow from combining the results given above.

5.5 *Failure and Ready Equivalence*

Let $\mathcal{P}$ be a PTP with state set S. A *failure set* of a stable state $s \in S$ with $s \not\longrightarrow$ is a set of external actions that cannot be carried out from s, i.e. $A \subseteq \mathsf{Act}$ is a failure set of s if $s \stackrel{a}{\not\twoheadrightarrow}$ for all $a \in A$. Clearly, if A is a failure set, so are all subsets of A. An appropriate testing scenario for *failure semantics* consists of a stochastic trace machine with an hourglass and action display as in the case of trace semantics but in addition for each $a \in \mathsf{Act}$ there is a switch which is used to block or unblock action a (see Figure 5.6 for an illustration). $\mathcal{P}$ can only perform free actions, i.e. actions that are currently not blocked. We call the machine the *stochastic failure machine*. After each visible action the experimenter can change the combination of free and blocked actions.
Assume that s is a stable state with $s \not\longrightarrow$ and the experiment is carried out under a total scheduler. The machine stagnates in state s if either s is a deadlock state or if the observer blocks all actions of the outgoing transitions of s. The failure set of s can be deduced from the set of actions that are not blocked by the observer.
Clearly, if $s \stackrel{\tau}{\twoheadrightarrow}$ or $s \longrightarrow$ the machine cannot halt in s and there exists no subset of Act that is a failure set of s. If the hourglass timer expires before the machine is forced to halt, a trace observation is recorded. Otherwise, a *failure observation* is made.

In this section, we do not consider time dependent schedulers but restrict to the class HR. We compare failure semantics with $=_{\mathsf{tr}}^{\mathsf{HR}}$ and forgo a comparison of failure semantics based on classes like $\mathsf{THR}, \mathsf{HD}, \mathsf{SR}, \ldots$ as this would not give any interesting new insights.

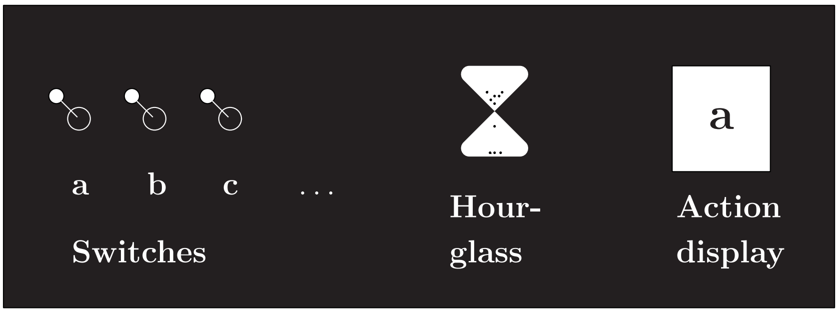

Figure 5.6: The stochastic failure machine.

Remark 5.5

For the experiment described above, we use two assumptions: We restrict to total schedulers to ensure that a deadlock is not caused by the scheduler (but by the set of blocked actions or by a deadlock state). Furthermore, we assume that in each step the scheduler always chooses a subdistribution over the set of free actions, i.e. if the observer chooses A as the set of free actions while $\mathcal{P}$ is in state s and ξ is the process history with $last(\xi) = s$, the choice $\mathcal{D}(\xi)$ of scheduler $\mathcal{D}$ is such that

$$\mathcal{D}(\xi)(a, \mu) > 0 \implies a \in A \cup \{\tau\}. \tag{5.6}$$

Formally, in the setting of failure semantics, observations are given by the following definition:

Definition 5.4 (Failure Observation)

Let $\mathcal{P}$ be a PTP and $\mathcal{D} \in \mathsf{HR}(\mathcal{P})$. A *failure observation* is a pair $(\mathsf{fa}_{\mathcal{P}}^{\mathcal{D}}, \mathsf{tr}_{\mathcal{P}}^{\mathcal{D}})$ where $\mathsf{tr}_{\mathcal{P}}^{\mathcal{D}}$ is the trace observation under $\mathcal{D}$ and $\mathsf{fa}_{\mathcal{P}}^{\mathcal{D}}$ is a function such that

$$\mathsf{fa}_{\mathcal{P}}^{\mathcal{D}} : \big(\mathsf{Act}^* \times \mathbb{R}_{\geq 0} \times \mathcal{P}(\mathsf{Act})\big) \to [0, 1]$$

and

$$\begin{aligned} \mathsf{fa}_{\mathcal{P}}^{\mathcal{D}}(\sigma, t, A) \quad = \quad & \mathsf{Pr}^{\mathcal{D}}(\{\pi \in \mathsf{path}(\mathcal{P}) \mid \pi \text{ is a } \mathcal{D}\text{-path}, \exists i : trace(\pi{\downarrow_i}) = \sigma, \\ & time(\pi{\downarrow_i}) \leq t, last(\pi{\downarrow_i}) \not\longrightarrow, \\ & \forall a \in A \cup \{\tau\} : last(\pi{\downarrow_i}) \overset{a}{\not\twoheadrightarrow} \}). \end{aligned}$$

The *set of failure observations* is defined as

$$\mathsf{O}_{\mathsf{fa}}(\mathcal{P}) = \{(\mathsf{tr}^{\mathcal{D}}_{\mathcal{P}}, \mathsf{fa}^{\mathcal{D}}_{\mathcal{P}}) \mid \mathcal{D} \text{ is an } \mathsf{HR}\text{-scheduler for } \mathcal{P}\}.$$

We claim that this definition characterizes the scenario of failure semantics as described above although both restrictions of Remark 5.5 are not used. Moreover, the set of free actions is only recorded for the last step. The following observations justify Definition 5.4:

- Assume that A is the set of free actions at a certain time instant but the machine is able to proceed with the execution of a visible action, say, a (probably preceded by a τ-transition). This means that the current state, say, s is either unstable or $s \xrightarrow{a}\!\!\!\!\rightarrow$ for some $a \in A$ and the knowledge of A does not give new insights in the communication capabilities of s.
- Assume that the scheduler causes a deadlock by choosing $\lambda^{\perp} = 1$. Definition 5.4 ignores the last choice of the scheduler but examines the actions the state reached by trace σ can perform. Thus, we neither have to restrict to total schedulers nor have to stipulate that Equation 5.6 holds.

Definition 5.5 (Failure Equivalence)
Two PTPs $\mathcal{P}_1$, $\mathcal{P}_2$ are *failure equivalent*, written $\mathcal{P}_1 =_{\mathsf{fa}} \mathcal{P}_2$, iff

$$\mathsf{O}_{\mathsf{fa}}(\mathcal{P}_1) = \mathsf{O}_{\mathsf{fa}}(\mathcal{P}_2).$$

Failure equivalence possesses more distinguishing power than trace equivalence. The same holds for ready equivalence. We summarize these results in Proposition 5.5 on page 111 after focusing on ready observations.

The *ready set* of a stable state s is the set of external actions that can be performed in s, i.e. $A \subseteq \mathsf{Act}$ is the ready set of s iff

$$A = \{a \in \mathsf{Act} \mid s \xrightarrow{a}\!\!\!\!\rightarrow\}.$$

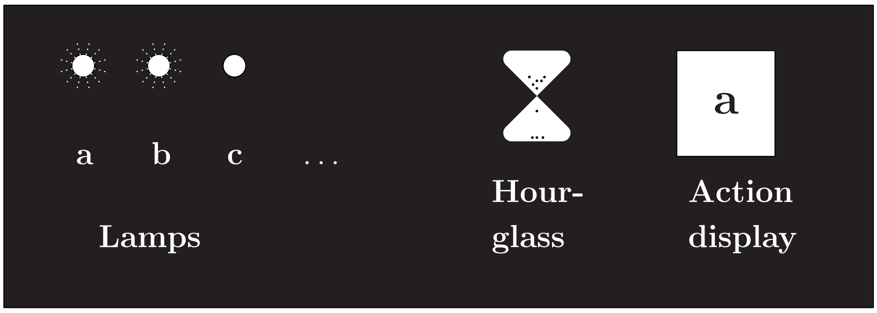

Figure 5.7: The stochastic ready machine

We define the following button pushing experiment[4] for ready semantics: The machine is the same as for trace semantics but additionally for each $a \in \mathsf{Act}$ there is a lamp. If the hourglass timer expires and the current state of the machine is s, the a-lamp is lit if and only if $s \xrightarrow{a}\!\!\!\!\rightarrow$. See Figure 5.7 for an illustration of the *stochastic ready machine*.

Definition 5.6 (Ready Observation)
Let $\mathcal{P}$ be a PTP and $\mathcal{D} \in \mathsf{HR}(\mathcal{P})$. A *ready observation* is a function

$$\mathsf{re}_{\mathcal{P}}^{\mathcal{D}} : \big(\mathsf{Act}^{*} \times \mathbb{R}_{\geq 0} \times \mathcal{P}(\mathsf{Act})\big) \rightarrow [0,1]$$

such that

$$\begin{aligned} \mathsf{re}_{\mathcal{P}}^{\mathcal{D}}(\sigma, t, A) \quad = \quad & \mathsf{Pr}^{\mathcal{D}}\big(\{\pi \in \mathsf{path}(\mathcal{P}) \mid \pi \text{ is a } \mathcal{D}\text{-path}, \exists i : \mathit{trace}(\pi{\downarrow_i}) = \sigma, \\ & \mathit{time}(\pi{\downarrow_i}) \leq t, \forall a \in \mathsf{Act} : (a \in A \iff \mathit{last}(\pi{\downarrow_i}) \xrightarrow{a}\!\!\!\!\rightarrow)\}\big). \end{aligned}$$

The *set of ready observations* is defined as

$$\mathsf{O}_{\mathsf{re}}(\mathcal{P}) = \{\mathsf{re}_{\mathcal{P}}^{\mathcal{D}} \mid \mathcal{D} \text{ is an } \mathsf{HR}\text{-scheduler for } \mathcal{P}\}.$$

As opposed to the definition of failure equivalence it is not necessary to

[4] We keep using the term “button pushing experiment” even if the machine has other features than buttons.

compare the trace observations in addition because of the equality

$$\sum_{A \subseteq \mathsf{Act}} \mathsf{re}_{\mathcal{P}}^{\mathcal{D}}(\sigma, t, A) = \mathsf{tr}_{\mathcal{P}}^{\mathcal{D}}(\sigma, t).$$

Definition 5.7 (Ready Equivalence)
Two PTPs $\mathcal{P}_1$, $\mathcal{P}_2$ are *ready equivalent*, written $\mathcal{P}_1 =_{\mathsf{re}} \mathcal{P}_2$, iff

$$\mathsf{O}_{\mathsf{re}}(\mathcal{P}_1) = \mathsf{O}_{\mathsf{re}}(\mathcal{P}_2).$$

In the probabilistic setting, a result which is analogous to the following proposition is given, for instance, in [vGla90, JS90].

Proposition 5.5
$=_{\mathsf{fa}}$ *and* $=_{\mathsf{re}}$ *are incomparable, but both are strictly finer than* $=_{\mathsf{tr}}^{\mathsf{tHR}}$.

Proof. The fact that $=_{\mathsf{fa}}$ and $=_{\mathsf{re}}$ are incomparable carries over from the nonprobabilistic setting [vGla90, Counterexample 6, Page 24]. To see the strictness consider the nonprobabilistic counterexamples in [vGla90] or compare [JS90] for the fully probabilistic case.
For the implication

$$\mathcal{P}_1 =_{\mathsf{fa}} \mathcal{P}_2 \implies \mathcal{P}_1 =_{\mathsf{tr}}^{\mathsf{tHR}} \mathcal{P}_2$$

first note that obviously $=_{\mathsf{fa}} \subset =_{\mathsf{tr}}^{\mathsf{HR}}$. Now, assume that there are PTPs $\mathcal{P}_1$ and $\mathcal{P}_2$ which are stochastic failure equivalent and $\mathcal{P}_1 \neq_{\mathsf{tr}}^{\mathsf{tHR}} \mathcal{P}_2$. From $\mathcal{P}_1 =_{\mathsf{fa}} \mathcal{P}_2$ we know that the observation of failure set Act are equal in P_1 and P_2. But Act is a failure set of a state s if and only if s is a deadlock state. This contradicts the assumption that deadlock states can be detected by $=_{\mathsf{tr}}^{\mathsf{tHR}}$. In a similar way, we can derive that $\mathcal{P}_1 =_{\mathsf{re}} \mathcal{P}_2$ implies $\mathcal{P}_1 =_{\mathsf{tr}}^{\mathsf{tHR}} \mathcal{P}_2$. □

Note that with a similar argumentation as for the proof of Proposition 5.2 (see page 90) we also derive that both, $=_{\mathsf{fa}}$ and $=_{\mathsf{re}}$, are strictly coarser than $=_{\mathsf{bs}}$.

5.6 Chapter Summary

For LTS-like models, trace semantics is known to be an important way to describe the behavior of a process: the visible parts of all execution paths the process can follow are listed. Nondeterministic choices are resolved deterministically based on the process history.
If probabilities come into play, we deal with a set of distributions which give the probability of a certain trace. In this setting, the resolution of nondeterminism is often associated to some kind of strategy related to an optimization problem (see, for instance, [Der70]). The process is considered as a closed system and constitutes a model for the analysis of this problem. This viewpoint leads to the definition of different scheduler classes.
In the timed setting, the current scheduler decision might depend on the time passed so far. Thus, even more scheduler classes are of interest.
We considered trace semantics with respect to a variety of scheduler classes and proved inclusions and also found out that many relations are incomparable. Additionally, we extended some well-established variants of trace equivalence to the setting of PTPs.

We do not expect any of the relations defined in this chapter to be a congruence with respect to the parallel composition of SPTPs since this is not even the case in the discrete-time setting (e.g. compare, for example, [Seg95]). For an example of two trace equivalent SPTPs which fail to be equivalent after composition we refer to Example 6.4 on page 132.

As an open problem, it remains to establish the relationship between $=_{\mathsf{tr}}^{\mathsf{SR}}$ and $=_{\mathsf{tr}}^{\mathsf{HR}}$. We have shown that $=_{\mathsf{tr}}^{\mathsf{HR}} \not\subset =_{\mathsf{tr}}^{\mathsf{SR}}$ but it is still unknown if $=_{\mathsf{tr}}^{\mathsf{SR}} \subset =_{\mathsf{tr}}^{\mathsf{HR}}$. For future work related to the topics of this section, we also mention axiomatizations and logical characterizations of the relations defined above.

Chapter 6

More button pushing experiments

6.1 Overview

In this chapter we retain the idea of button pushing experiments. An experiment is carried out in which the process is represented by a black box. As before, during a run of the process the visible process behavior is shown at a display and the experimenter determines which external stimuli are given to the process. But as opposed to the previous chapter, here a PTP is considered as an open system which is strongly influenced by its environment. Since visible actions constitute the interface to the outside world, external nondeterminism is no longer resolved arbitrarily such as in the case of trace equivalence. Instead, two processes are regarded as equivalent if they have equivalent trace observations under fixed environment conditions. Thus, we compare their behavior provided that the *same* external stimuli are given. Stated informally, $\mathcal{P}$ is equivalent to $\mathcal{Q}$ if for all environments E

$$\text{Observations}(\mathcal{P} \text{ operates in } E) = \text{Observations}(\mathcal{Q} \text{ operates in } E).$$

There are several ways to represent a communication environment of a PTP. One possibility is to maintain the scheduler classes $\{\mathsf{THR}, \mathsf{THD}, \mathsf{HR}, \ldots\}$ of Definition 3.5 (see page 48) and to relate certain schedulers, for example, if scheduler $\mathcal{D}$ for PTP $\mathcal{P}$ resolves external nondeterminism in the same way as scheduler $\mathcal{D}'$ for PTP $\mathcal{Q}$ does. Another possibility is to follow some well-established approaches from the nonprobabilistic setting. There, the idea is that the environment of a process is simulated by assuming that in each step only a certain subset of visible actions is offered by the environment. In that context the observations are called failure traces [Phi87] or ready traces [Pnu85, BB87]. Here, we extend these ideas to the stochastic setting. The extension presented in the sequel differs from trace semantics in the following points:

- Choices resolving external nondeterminism are not made on the basis of the *complete* process history but on its visible part only.
- Processes are tested under fixed environment conditions in order to decide if they should be related or not.

In the probabilistic setting, a further distinction regarding the environment of a process can be made. The "nature" of the environment can be either completely nonprobabilistic or probabilistic. We show that assuming a probabilistic environment increases the distinguishing power of the resulting notions of equivalence.
In the setting of this thesis, processes act in continuous time which gives rise to a further distinction: Is the environment aware of time or are external stimuli given in a time-abstract manner?
We take the above considerations into account by proceeding as follows: Similar to trace semantics, a button pushing experiment is used to describe our approach in which actions are externally available either a) with a certain probability or b) after a certain delay. The resulting semantics are called *weighted trace semantics* in the case of a) and *delayed trace semantics* for scenario b).

All relations presented in the sequel are analyzed with respect to parallel composition of SPTPs. It turns out that even relations based on the simulation of a probabilistic environment are not powerful enough to ensure the congruence property (see Counterexample 6.4 on page 132). And also in case that external stimuli are given after a certain duration, we found out that the resulting relation does not have the congruence property (see Counterexample 6.7 on page 148).

Weighted Traces The *weighted trace machine*[1] (as illustrated in Figure 6.1 on page 117) is equipped with *controllers*, one for each action, which are used to resolve nondeterministic choices between the different actions the PTP under study can execute. The observer can adjust the controllers to determine the probability at which a certain action is (immediately) available. New adjustments can be made when a visible action is executed by the process. As a by-product, we present a notion of *internal schedulers* which are restricted to the resolution of internal nondeterminism.

Since the weighted trace machine has more features to "test" a process we cannot expect any of the trace equivalences of Chapter 5 to imply *weighted trace equivalence*. The opposite direction also does not hold: We show that weighted trace equivalence is not finer than trace equivalence (and thus not finer than ready or failure equivalence) but for a good reason: The experimenter of the weighted trace scenario has not access to the complete process history (but only to the trace performed so far). Her choice for the resolution of external nondeterminism is based on the visible part of the process history (whereas the choice of the internal scheduler depends on the complete history). As opposed to that, in the case of trace semantics, an HR-scheduler can have different choices after path fragments with the same trace. We elucidate this relationship by defining *trace dependent schedulers* and show that the resulting trace equivalence is strictly finer than weighted trace equivalence.

[1] To improve readability, we drop the adjective "stochastic" for the machines considered in this chapter.

Delayed Traces The delayed traces approach relies on the idea that the environment provides external stimuli after a certain delay, i.e. the environment is of a "timed nature". The corresponding *delayed trace machine* is as the trace machine for PTPs but additionally equipped with *(action) count-down timers*, one for each visible action (compare Figure 6.9 on page 137). The process can communicate with the environment via an action, say, a, after the timer of a has expired. The observer determines the respective delays by setting the action timers. It is possible to choose the zero delay for an action meaning that this action is immediately available. If more than one timer is set to zero, the remaining nondeterminism has to be resolved by a scheduler.

It turns out that *delayed trace equivalence* fails to be comparable to the trace equivalences defined in the previous chapter for the same reason as weighted trace equivalence does so. Surprisingly, it is also not comparable to weighted trace equivalence which is due to the fact that

- a delay imposed by the environment increases the distinguishing power of the machine and
- it may be the case that several actions are immediately externally available and this external nondeterminism is not resolved by the experimenter (which is never the case in the scenario of weighted traces).

It is important to point out that the delayed traces scenario describes a new class of schedulers. We did not list schedulers which make their decision after a certain delay in the scheduler definition of Chapter 3 (see Definition 3.6 on page 50). Instead we present this idea here in form of a enriched button pushing experiment to ensure that while comparing two processes fixed environment conditions are used.

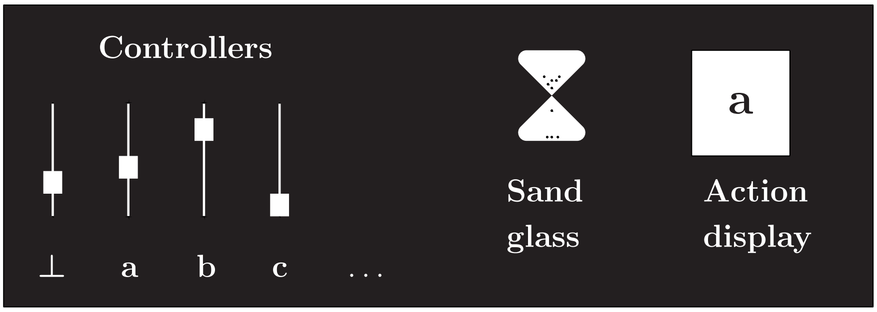

Figure 6.1: The weighted trace machine

6.2 *Weighted Trace Equivalence*

The button pushing experiment associated with *weighted trace semantics* is as follows (compare Figure 6.1): As for trace semantics the black box simulator has an action display and an hourglass. Additionally, there are *controllers*, one for each visible action and one labeled by $\perp$. The experimenter can adjust the controllers in order to determine the probability at which the corresponding action is supplied by the environment. More precisely, each controller position specifies a weight which equals the probability of the associated action after normalization. The $\perp$-controller represents the probability that no external stimulus is given. The observer can make new controller adjustments after the execution of a visible action. This means that the weight of an action stays constant until the process performs the next visible action.

In the setting of failure/ready traces, actions are either blocked (available with probability zero) or free (available with probability one). Since PTPs branch probabilistically, it is appropriate to assume that in our setting the observer can decide that with a certain probability an action is immediately available. The controller adjustments during the whole experiment can be described by a function

$$\mathsf{ctr}: \underbrace{\mathsf{Act}^*}_{\text{interaction history}} \times \underbrace{\big(\mathsf{Act} \cup \{\bot\}\big)}_{\text{controller}} \rightarrow \underbrace{\mathbb{R}_{\geq 0}}_{\text{controller position}} .$$

Let $\tilde{\sigma} \in \mathsf{Act}^*$, $a \in \mathsf{Act}$ and let $\mathcal{P}$ be the PTP under study. If $\mathsf{ctr}_{\tilde{\sigma}}(a) := \mathsf{ctr}(\tilde{\sigma}, a) > 0$, action a is immediately provided by the environment after trace $\tilde{\sigma}$. If a is possible in the current state of $\mathcal{P}$ the probability that a is executed by $\mathcal{P}$ is $\mathsf{ctr}_{\tilde{\sigma}}(a)/N$ where N is a normalization factor. In case that $\mathsf{ctr}_{\tilde{\sigma}}(a) = 0$, action a is completely blocked. If $\mathsf{ctr}_{\tilde{\sigma}}(\bot) > 0$, there is a non-zero probability that the environment refuses any interaction with the process.

Obviously, for a given scheduler the underlying probability measure of paths changes if communication opportunities are limited according to ctr. We assume that the remaining internal nondeterminism is resolved by a scheduler and give a new definition which elucidates the role a scheduler has in the setting of weighted trace semantics. Let $\mathcal{Z}$ be the set of functions

$$\vartheta : (\mathsf{Act}_\tau \times \mathsf{dis}_S) \rightarrow [0, 1]$$

such that for all $a \in \mathsf{Act}_\tau$ we have that $\vartheta(a, \cdot) =: \vartheta_a$ is a subdistribution on dis_S. A scheduler can choose ϑ_a to resolve the internal nondeterminism which occurs if a state has several outgoing transitions labeled by a.

Definition 6.1 (Internal Scheduler)
A (time independent) *internal scheduler* for PTP P is a function

$$\mathcal{E} : \mathsf{pathf}(\mathcal{P}) \rightarrow \mathcal{Z}$$

such that for $\xi \in \mathsf{pathf}(\mathcal{P})$ with $last(\xi) = s$ and $\mathcal{E}(\xi) =: \vartheta$ the following conditions hold:

i) $\vartheta(a,\mu) > 0$ implies $s \xrightarrow{a} \mu$.

ii) Whenever s is stable, ϑ_a is a distribution for all $a \in \mathsf{Act}$ with $s \xrightarrow{a}$.

iii) Whenever s is unstable, ϑ_τ is a distribution.

iv) For all path fragments $\xi, \xi' \in \mathsf{pathf}(\mathcal{P})$,
if $untime(\xi) = untime(\xi')$ then $\mathcal{E}(\xi) = \mathcal{E}(\xi')$.

Intuitively, the third condition ensures that if s is unstable, $\vartheta_a^\perp$ can be non-zero for some $a \in \mathsf{Act}$. Thus, it enables the internal scheduler to decide (with a non-zero probability) for an invisible transition but for a. We explain this more detailed in the sequel.
Assume that $\mathcal{P}$ is simulated by the weighted trace machine and $\xi \in \mathsf{pathf}(\mathcal{P})$. The controller positions in state $last(\xi) = s$ are given by $\mathsf{ctr}_{\tilde\sigma}$ where $\tilde\sigma = trace(\xi)$. If s is stable and $\mathsf{ctr}_{\tilde\sigma}(a) > 0$ then transition $s \xrightarrow{a} \mu$ is (immediately) taken with probability $\vartheta(a,\mu) \cdot \mathsf{ctr}_{\tilde\sigma}(a)/N(\xi,\mathsf{ctr})$ under internal scheduler $\mathcal{E}$ with $\mathcal{E}(\xi) = \vartheta$. The normalization factor $N(\xi,\mathsf{ctr})$ is given by

$$N(\xi,\mathsf{ctr}) = \sum_{\substack{a \in \mathsf{Act}: s \xrightarrow{a}, \\ \mathsf{ctr}_{\tilde\sigma}(a) \in \mathbb{R}_{>0}}} \mathsf{ctr}_{\tilde\sigma}(a) + \mathsf{ctr}_{\tilde\sigma}(\perp). \tag{6.1}$$

The probability of leaving s *not* immediately equals the probability that the environment refuses an immediate interaction. This happens with probability

$$\mathsf{soj}(\xi,\mathsf{ctr}) := \mathsf{ctr}_{\tilde\sigma}(\perp)/N(\xi,\mathsf{ctr})$$

if $N(\xi,\mathsf{ctr}) > 0$ and $\mathsf{soj}(\xi,\mathsf{ctr}) := 1$ otherwise (soj stands for "sojourn"). This probability is determined by the observer and initiates a race between the outgoing PH transitions of s. If immediate interaction is permitted by the observer, the internal scheduler cannot inhibit this.

If s is unstable, we put $\mathsf{soj}(\xi,\mathsf{ctr}) := 0$. In case that $\mathsf{ctr}_{\tilde\sigma}(a) > 0$ and $a \neq \tau$ transition $s \xrightarrow{a} \mu$ is taken with probability $\vartheta(a,\mu) \cdot \mathsf{ctr}_{\tilde\sigma}(a)/N(\xi,\mathsf{ctr})$. However, the probability that an invisible transition is taken is the sum of p_e and p_i where p_e is the probability that the environment refuses any interaction and p_i is the probability that scheduler $\mathcal{E}$ decides against a visible transition. More precisely,

$$p_e := \begin{cases} \frac{\mathsf{ctr}_{\tilde{\sigma}}(\perp)}{N(\xi,\mathsf{ctr})} & \text{if } N(\xi,\mathsf{ctr}) > 0, \\ 1 & \text{otherwise,} \end{cases}$$
$$p_i := \begin{cases} 1 - p_e & \text{if } \nexists a : s \xrightarrow{a}\!\!\!\!\rightarrow, \mathsf{ctr}_{\tilde{\sigma}}(a) > 0, \\ & \text{or } N(\xi,\mathsf{ctr}) = 0, \\ \sum\limits_{\substack{a \in \mathsf{Act}: s \xrightarrow{a}\!\!\!\rightarrow, \\ \mathsf{ctr}_{\tilde{\sigma}}(a) \in \mathbb{R}_{>0}}} \vartheta_a^{\perp} \cdot \frac{\mathsf{ctr}_{\tilde{\sigma}}(a)}{N(\xi,\mathsf{ctr})} & \text{otherwise.} \end{cases} \tag{6.2}$$

Thus, transition $s \xrightarrow{\tau}\!\!\!\!\rightarrow \mu$ has probability $\vartheta(\tau,\mu) \cdot (p_i + p_e)$. This definition is reasonable because with probability p_e the environment refuses any interaction. Thus, the probability of a τ-transition must be greater or equal to p_e. On the other hand, $\mathcal{E}$ can always decide for τ, independent of $\mathsf{ctr}_{\tilde{\sigma}}$.

Next, we focus on the definition of a probability measure on sets of paths that are executed while nondeterministic branching is resolved by ctr and $\mathcal{E}$. A $(\mathsf{ctr},\mathcal{E})$-*path* is a path

$$\pi = s_1 \xrightarrow{\mathsf{e}_1} s_2 \xrightarrow{\mathsf{e}_2} \ldots \in \mathsf{path}(\mathcal{P})$$

that respects the decisions of ctr and $\mathcal{E}$, i.e., for all $i \geq 1$ there exists $\mu_i \in \mathsf{dis}_S$ with $\mu_i(s_{i+1}) > 0$ and whenever

$$\mathsf{e}_i = \begin{cases} t \in \mathbb{R}_{>0} & \text{then } \mathsf{soj}(\xi,\mathsf{ctr}) > 0, \\ a \in \mathsf{Act} & \text{then } \mathsf{ctr}_{\tilde{\sigma}}(a) \in \mathbb{R}_{>0} \text{ and } \vartheta(a,\mu_i) > 0, \\ \tau & \text{then } \vartheta(\tau,\mu_i) > 0 \text{ and } p_e + p_i > 0, \end{cases}$$

where $trace(\pi{\downarrow_i}) = \tilde{\sigma}$, $\mathcal{D}(\pi{\downarrow_i}) = \vartheta$ and p_e, p_i are as defined above. A $(\mathsf{ctr},\mathcal{E})$-path fragment is a path fragment of a $(\mathsf{ctr},\mathcal{E})$-path.

The pair $(\mathsf{ctr},\mathcal{E})$ induces a probability space over sample set Ω which is the set of all $(\mathsf{ctr},\mathcal{E})$-paths. A sigma-algebra $\Sigma^{(\mathsf{ctr},\mathcal{E})}$ is constructed in a similar way as in Section 3.6 for THR-schedulers. The basis is the set $\mathcal{C}^{(\mathsf{ctr},\mathcal{E})}$ of cylinder sets of $(\mathsf{ctr},\mathcal{E})$-paths. Probability measure $\mathsf{Pr}^{(\mathsf{ctr},\mathcal{E})}$ on $\Sigma^{(\mathsf{ctr},\mathcal{E})} := \sigma(\mathcal{C}^{(\mathsf{ctr},\mathcal{E})})$ is defined by specifying the probabilities for the elements of $\mathcal{C}^{(\mathsf{ctr},\mathcal{E})}$. Let

$$\zeta = s_1\ E_1\ s_2\ E_2 \dots\ E_{k-1}\ s_k$$

where $s_1, s_2, \dots, s_k \in S$ and for $i \in \{1, 2, \dots, k-1\}$ either $E_i = \{a\}$ for some $a \in \mathsf{Act}_\tau$ or $E_i = (x, y] \subseteq \mathbb{R}_{>0}$, $x < y$. Then cylinder set $\mathsf{C}_\zeta \in \mathcal{C}^{(\mathsf{ctr},\mathcal{E})}$ contains all $(\mathsf{ctr},\mathcal{E})$-paths

$$\pi = s'_1 \xrightarrow{\mathsf{e}_1} s'_2 \xrightarrow{\mathsf{e}_2} \dots$$

such that $s_j = s'_j$, $1 \leq j \leq k$ and $\mathsf{e}_i \in E_i$ for all $i \in \{1, 2, \dots, k-1\}$.
For $k = 1$ we define

$$\mathsf{Pr}^{(\mathsf{ctr},\mathcal{E})}(\mathsf{C}_s) = \nu(s)$$

where ν is the initial distribution of $\mathcal{P}$. Now, let $\mathsf{C}_\zeta \in \mathcal{C}^{(\mathsf{ctr},\mathcal{E})}$, $|\zeta| > 1$ and $\pi \in \mathsf{C}_\zeta$, $\xi := \pi{\downarrow_{k-1}}$. The controller positions are given by $\mathsf{ctr}_{\tilde{\sigma}}$ where $trace(\xi) = \tilde{\sigma}$ and the decision of internal scheduler $\mathcal{E}$ is denoted by $\mathcal{E}(\xi) = \vartheta$. Let

$$\zeta' = s_1\ E_1\ s_2\ E_2 \dots\ E_{k-2}\ s_{k-1}.$$

We distinguish three cases:

1. In the case that E_{k-1} is an interval, we know that s_{k-1} is stable and define

   $$\mathsf{Pr}^{(\mathsf{ctr},\mathcal{E})}(\mathsf{C}_\zeta) := \mathsf{Pr}^{(\mathsf{ctr},\mathcal{E})}(\mathsf{C}_{\zeta'}) \cdot \mathsf{soj}(\xi, \mathsf{ctr}) \cdot \mathsf{reach}_{s_{k-1}}(\{s_{k-1}\}, \{s_k\}, E_{k-1}).$$

   Recall that $\mathsf{soj}(\xi, \mathsf{ctr})$ is the probability of not leaving $last(\xi)$ immediately and $\mathsf{reach}_s(\{s\}, \{s'\}, E_{k-1})$ is the one-step probability to reach state s' from s at time instant $t \in E_{k-1}$ (compare Section 3.3, page 41).

2. Assume that $E_{k-1} = \{a\}, a \in \mathsf{Act}$. This implies that $s_{k-1} \xrightarrow{a}\!\!\!\!\rightarrow$ and $\mathsf{ctr}_{\tilde{\sigma}}(a) \in \mathbb{R}_{>0}$. The observer decides for a with probability $\frac{\mathsf{ctr}_{\tilde{\sigma}}(a)}{N(\xi,\mathsf{ctr})}$ and $\mathcal{E}$ chooses transition (a, μ) with probability $\vartheta(a, \mu)$. Therefore

   $$\mathsf{Pr}^{(\mathsf{ctr},\mathcal{E})}(\mathsf{C}_\zeta) := \mathsf{Pr}^{(\mathsf{ctr},\mathcal{E})}(\mathsf{C}_{\zeta'}) \cdot \frac{\mathsf{ctr}_{\tilde{\sigma}}(a)}{N(\xi,\mathsf{ctr})} \cdot \sum \left\{\!\!\left| \vartheta(a, \mu) \cdot \mu(s_k) \mid \exists \mu : s_{k-1} \xrightarrow{a}\!\!\!\!\rightarrow \mu \right|\!\!\right\}.$$

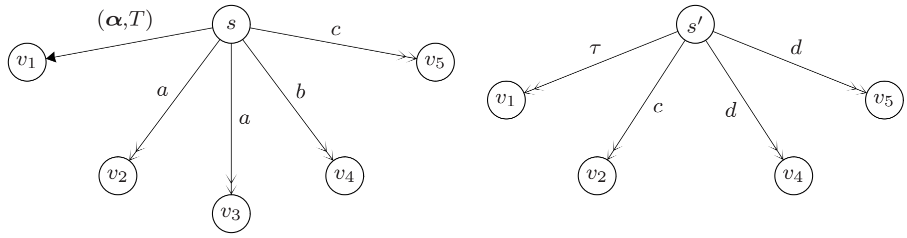

Figure 6.2: Outgoing transitions of state s and s' of PTP $\mathcal{P}$

3. Suppose that $E_{k-1} = \{\tau\}$. As already explained above, τ can be performed if either the environment refuses instantaneous interaction (which happens with probability p_e) or if $\mathcal{E}$ decides against a visible transition (probability p_i). Let p_e and p_i be defined as above. Then

$$\mathsf{Pr}^{(\mathsf{ctr},\mathcal{E})}(\mathsf{C}_\zeta) := \mathsf{Pr}^{(\mathsf{ctr},\mathcal{E})}(\mathsf{C}_{\zeta'}) \cdot (p_i + p_e) \cdot \sum \left\{\!\left| \vartheta(\tau, \mu) \cdot \mu(s_k) \mid \exists \mu : s_{k-1} \xrightarrow{\tau} \mu \right|\!\right\}.$$

Similar to the construction in Section 3.6, measure $\mathsf{Pr}^{(\mathsf{ctr},\mathcal{E})}$ can be extended to a unique probability measure on the complete sigma-algebra $\Sigma^{(\mathsf{ctr},\mathcal{E})}$. Sometimes, we may take as sample space the set of all paths instead of the set of $(\mathsf{ctr}, \mathcal{E})$-paths by assuming that sets of paths that are prohibited by ctr or $\mathcal{E}$ have probability zero. Moreover, we may write $\mathsf{Pr}_{\mathcal{P}}^{(\mathsf{ctr},\mathcal{E})}$ instead of $\mathsf{Pr}^{(\mathsf{ctr},\mathcal{E})}$.

Example 6.1
Suppose that PTP $\mathcal{P}$ is analyzed with respect to weighted traces. For simplicity, we only consider a single step in $\mathcal{P}$. Let s (s' respectively) be the a state of $\mathcal{P}$ and assume that this state is entered at time instant t after trace $\tilde{\sigma}$ is performed. The transitions of s and s' are depicted in Figure 6.2 (for simplicity we assume that all target distributions are Dirac distributions). We consider controller adjustments $\mathsf{ctr}_{\tilde{\sigma}}$ with values given by the following table:

Action	a	b	c	d	$\bot$
$\mathsf{ctr}_{\tilde{\sigma}}$	5	2	0	2	3

An internal scheduler $\mathcal{E}$ is used to resolve occurring internal nondeterminism. Let ξ with $last(\xi) = s$ and $trace(\xi) = \tilde{\sigma}$ (ξ' with $last(\xi') = s'$ and $trace(\xi') = \tilde{\sigma}$, respectively) be a path fragment of $\mathcal{P}$. The scheduler decision is given by $\mathcal{E}(\xi) = \vartheta$ and $\mathcal{E}(\xi') = \vartheta'$ where ϑ and ϑ' are defined as follows:

Transition		(s, a, δ_{v_2})	(s, a, δ_{v_3})	(s, b, δ_{v_4})	(s, c, δ_{v_5})
Scheduler	ϑ	1/2	1/2	1	1

Transition		(s', τ, δ_{v_1})	(s', c, δ_{v_2})	(s', d, δ_{v_4})	(s', d, δ_{v_5})
Scheduler	ϑ'	1	2/3	1/4	0

Note that $\vartheta_a, \vartheta_b, \vartheta_c$ are distributions because s is stable. Obviously, s is left immediately with probability 7/10 if the controllers are positioned according to $\mathsf{ctr}_{\tilde{\sigma}}$. Thus, PH transition $(s, (\boldsymbol{\alpha}, T), \delta_{v_1})$ is taken with probability 3/10. Let us consider the probability to reach state v_2 from s after history ξ. If $\pi{\downarrow_i} = \xi$ then

$$\begin{aligned} & \mathsf{Pr}^{(\mathsf{ctr},\mathcal{E})}(\{\pi \mid \pi{\downarrow_i} = \xi \text{ and } last(\pi{\downarrow_{i+1}}) = v_2\}) \\ = \; & \mathsf{Pr}^{(\mathsf{ctr},\mathcal{E})}(\{\pi \mid \pi{\downarrow_i} = \xi\}) \cdot 5/10 \cdot 1/2 \end{aligned}$$

since v_2 can only be reached by an a-transition. The probability at which a is externally available is 5/10 and $\mathcal{E}$ chooses (s, a, δ_{v_2}) with probability 1/2. For the probability to reach v_1 within $x > 0$ time units we calculate

$$\begin{aligned} & \mathsf{Pr}^{(\mathsf{ctr},\mathcal{E})}(\{\pi \mid \pi{\downarrow_i} = \xi, last(\pi{\downarrow_{i+1}}) = v_1, time(\pi{\downarrow_{i+1}}) \leq x + time(\pi{\downarrow_i}))\}) \\ = \; & \mathsf{Pr}^{(\mathsf{ctr},\mathcal{E})}(\{\pi \mid \pi{\downarrow_i} = \xi\}) \cdot 3/10 \cdot \mathsf{reach}_s(\{s\}, \{v_1\}, [0, x]). \end{aligned}$$

We now treat s'. Unstable states are left immediately, regardless of the controller settings. In the case of scheduler choice ϑ' and path fragment ξ' with $\pi{\downarrow_i}= \xi'$, state v_1 is reached with probability

$$\begin{aligned} & \mathsf{Pr}^{(\mathsf{ctr},\mathcal{E})}(\{\pi \mid \pi{\downarrow_i}= \xi' \text{ and } last(\pi{\downarrow_{i+1}}) = v_1\}) \\ = \; & \mathsf{Pr}^{(\mathsf{ctr},\mathcal{E})}(\{\pi \mid \pi{\downarrow_i}= \xi'\}) \cdot (p_e + p_i) \cdot 1 \\ = \; & \mathsf{Pr}^{(\mathsf{ctr},\mathcal{E})}(\{\pi \mid \pi{\downarrow_i}= \xi'\}) \cdot (3/5 + 2/5 \cdot 3/4). \end{aligned}$$

Definition 6.2 (Weighted Trace Observation)
Let $\mathsf{ctr} : \mathsf{Act}^* \times (\mathsf{Act} \cup \{\bot\}) \to \mathbb{R}_{\geq 0}$ describe the controller positions and let $\mathcal{E}$ be an internal scheduler for PTP $\mathcal{P}$. A *weighted trace observation* is a function

$$\mathsf{we}_{\mathcal{P}}^{(\mathsf{ctr},\mathcal{E})} : \big(\mathsf{Act}^* \times \mathbb{R}_{\geq 0}\big) \to [0,1]$$

such that

$$\begin{aligned} \mathsf{we}_{\mathcal{P}}^{(\mathsf{ctr},\mathcal{E})}(\sigma, t) \; = \; & \mathsf{Pr}_{\mathcal{P}}^{(\mathsf{ctr},\mathcal{E})}(\{\pi \in \mathsf{path}(\mathcal{P}) \mid \pi \text{ is a } (\mathsf{ctr},\mathcal{E})\text{-path}, \\ & \exists i : trace(\pi{\downarrow_i}) = \sigma, time(\pi{\downarrow_i}) \leq t\}). \end{aligned}$$

The *set of weighted trace observations* of $\mathcal{P}$ with respect to ctr is defined as

$$\mathsf{O}_{\mathsf{we}}^{\mathsf{ctr}}(\mathcal{P}) = \{\mathsf{we}_{\mathcal{P}}^{(\mathsf{ctr},\mathcal{E})} \mid \mathcal{E} \text{ is an internal scheduler for } \mathcal{P}\}.$$

Definition 6.3 (Weighted Trace Equivalence)
Two PTPs $\mathcal{P}_1$ and $\mathcal{P}_2$ are *weighted trace equivalent*, written $\mathcal{P}_1 =_{\mathsf{we}} \mathcal{P}_2$, iff for all functions $\mathsf{ctr} : \mathsf{Act}^* \times (\mathsf{Act} \cup \{\bot\}) \to \mathbb{R}_{\geq 0}$

$$\mathsf{O}_{\mathsf{we}}^{\mathsf{ctr}}(\mathcal{P}_1) = \mathsf{O}_{\mathsf{we}}^{\mathsf{ctr}}(\mathcal{P}_2).$$

Our next objective is the relationship between $=_{\mathsf{we}}$ and failure trace equivalence [Phi87]. We define *failure trace observations* as a special case of weighted trace observations:

Definition 6.4 (Failure Trace Equivalence)
Two PTPs $\mathcal{P}_1$ and $\mathcal{P}_2$ are *failure trace equivalent*, written $\mathcal{P}_1 =_{\mathsf{ftr}} \mathcal{P}_2$, iff for all functions $\mathsf{ctr} : \mathsf{Act}^* \times (\mathsf{Act} \cup \{\perp\}) \to \mathbb{R}_{\geq 0}$ and all internal schedulers $\mathcal{E}_1$ for $\mathcal{P}_1$ there exists an internal scheduler $\mathcal{E}_2$ for $\mathcal{P}_2$ such that

$$\lim_{t\to\infty} \mathsf{we}_{\mathcal{P}_1}^{(\mathsf{ctr},\mathcal{E}_1)}(\sigma, t) > 0 \iff \lim_{t\to\infty} \mathsf{we}_{\mathcal{P}_2}^{(\mathsf{ctr},\mathcal{E}_2)}(\sigma, t) > 0.$$

We only distinguish, if σ can be performed under the pair $(\mathsf{ctr}, \mathcal{E})$ or not. We do not compare the probabilities nor set time limits.

Proposition 6.1
Let $\mathcal{P}_1$ and $\mathcal{P}_2$ be PTP*s. Then $\mathcal{P}_1 =_{\mathsf{we}} \mathcal{P}_2$ implies $\mathcal{P}_1 =_{\mathsf{ftr}} \mathcal{P}_2$.*

Obviously, the opposite direction of the above proposition does not hold. This can be seen by considering a similar example as in the case of trace equivalence and probabilistic trace equivalence (compare Example 5.1 on page 88).

For LTSs it was shown that failure trace equivalence does neither imply ready nor failure equivalence (see [vGla90, Page 21]). The following example proves a similar result in the setting of PTPs: weighted trace equivalence neither implies failure nor ready equivalence. Because of the effect the controller positions have on the resolution of external nondeterminism and the fact that we consider randomized schedulers, the argumentation is more complicated than in the nonprobabilistic case.

Example 6.2
Consider the two PTPs in Figure 6.3. It is easily seen that $\mathcal{P}$ and $\mathcal{Q}$ are failure and ready equivalent (see Definition 5.5 and 5.7 on page 109 and 111) and therefore also $\mathcal{P} =_{\mathsf{tr}}^{\mathsf{HR}} \mathcal{Q}$. But $\mathcal{P} \neq_{\mathsf{we}} \mathcal{Q}$. This can be seen by assuming that the observer

1. initially allows the immediate execution of action a by setting $\mathsf{ctr}_\epsilon(a) > 0$,

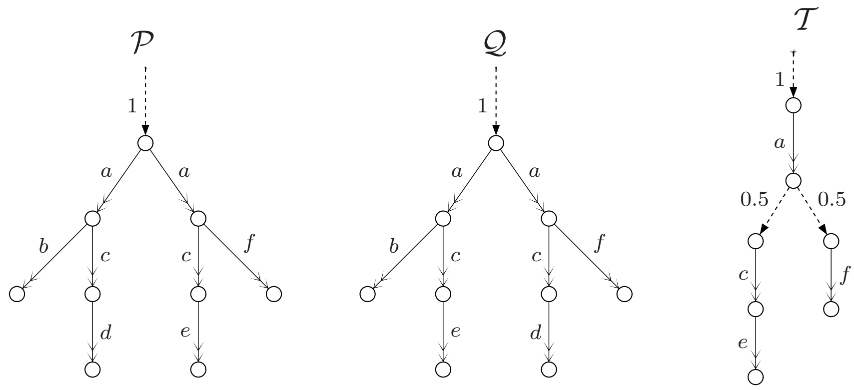

Figure 6.3: $\mathcal{P} =_{\mathsf{fa}} \mathcal{Q}$ and $\mathcal{P} =_{\mathsf{re}} \mathcal{Q}$, but $(\mathcal{P}, \mathcal{Q}) \notin =_{\mathsf{we}}, =_{\mathsf{de}}, =_{\mathsf{te}}$.

2. assigns positive weights only to c and f in the next step, i.e. $\mathsf{ctr}_a(c) = \mathsf{ctr}_a(f) = 1$ and $\mathsf{ctr}_a(b) = 0$,

3. blocks all actions except e completely in the final step (after trace ac has been performed), i.e. $\mathsf{ctr}_{ac}(e) = 1$ and $\mathsf{ctr}_{ac}(d) = 0$.

Furthermore, we assume that the $\perp$-controller is at position 0 during the whole experiment. Note that internal nondeterminism is only resolved in the first step. If now internal scheduler $\mathcal{E}_{\mathcal{P}}$ for $\mathcal{P}$ chooses the right a-transition with probability one, we have for all $t \geq 0$

$$\begin{array}{lllllll} \mathsf{we}_{\mathcal{P}}^{(\mathsf{ctr}, \mathcal{E}_{\mathcal{P}})}(ace, t) & = & \mathsf{we}_{\mathcal{P}}^{(\mathsf{ctr}, \mathcal{E}_{\mathcal{P}})}(ac, t) & = & \mathsf{we}_{\mathcal{P}}^{(\mathsf{ctr}, \mathcal{E}_{\mathcal{P}})}(af, t) & = & 1/2, \\ \mathsf{we}_{\mathcal{P}}^{(\mathsf{ctr}, \mathcal{E}_{\mathcal{P}})}(a, t) & = & 1. \end{array}$$

An internal scheduler $\mathcal{E}_{\mathcal{Q}}$ for $\mathcal{Q}$ must choose the left a-transition with probability 1/2 because only then $\mathsf{we}_{\mathcal{P}}^{(\mathsf{ctr}, \mathcal{D})}(ace, t) = \mathsf{we}_{\mathcal{Q}}^{(\mathsf{ctr}, \mathcal{D}')}(ace, t)$ holds. Since the choice of an internal scheduler is a distribution on the set of equally labeled transitions, the right a-transition is also chosen with probability 1/2. We get for all $t \geq 0$

$$\begin{array}{llllll} \mathsf{we}_{\mathcal{Q}}^{(\mathsf{ctr}, \mathcal{D}')}(ace, t) & = & 1/2, & \mathsf{we}_{\mathcal{Q}}^{(\mathsf{ctr}, \mathcal{D}')}(ac, t) & = & 3/4, \\ \mathsf{we}_{\mathcal{Q}}^{(\mathsf{ctr}, \mathcal{D}')}(af, t) & = & 1/4, & \mathsf{we}_{\mathcal{Q}}^{(\mathsf{ctr}, \mathcal{D}')}(a, t) & = & 1. \end{array}$$

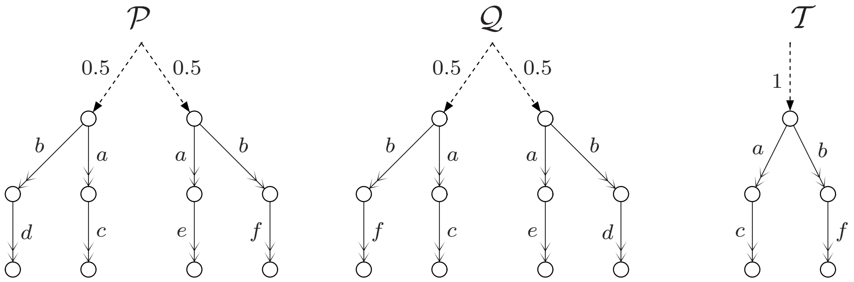

Figure 6.4: $(\mathcal{P}, \mathcal{Q}) \notin =^{\mathsf{HR}}_{\mathsf{tr}}, =_{\mathsf{de}}, =_{\mathsf{te}}$ but $\mathcal{P} =_{\mathsf{we}} \mathcal{Q}$

Obviously, a different resolution of the initial nondeterministic choice in $\mathcal{Q}$ does also not lead to a matching observation. Thus, there is no internal scheduler $\mathcal{E}_{\mathcal{Q}}$ for $\mathcal{Q}$ that can match this combination.

The next example proves that weighted trace equivalence does not imply failure and ready equivalence. We summarize the results of the two counterexamples in Proposition 6.2 and give an informal explanation of the incomparableness.

Example 6.3
The two PTPs in Figure 6.4 are not in $=^{\mathsf{HR}}_{\mathsf{tr}}$ (and therefore also neither failure nor ready equivalent). To see this, assume that HR-scheduler $\mathcal{D}$ for $\mathcal{P}$ is such that $\mathsf{tr}^{\mathcal{D}}_{\mathcal{P}}(ac, t) = \mathsf{tr}^{\mathcal{D}}_{\mathcal{P}}(bf, t) = 0.5$ and $\mathsf{tr}^{\mathcal{D}}_{\mathcal{P}}(ae, t) = \mathsf{tr}^{\mathcal{D}}_{\mathcal{P}}(bd, t) = 0$ (for all $t \geq 0$). Now, an HR-scheduler $\mathcal{D}'$ for $\mathcal{Q}$ must choose the left a-transition with probability one to match $\mathsf{tr}^{\mathcal{D}}_{\mathcal{P}}(ac, t) = \mathsf{tr}^{\mathcal{D}'}_{\mathcal{Q}}(ac, t) = 0.5$. But then $\mathsf{tr}^{\mathcal{D}'}_{\mathcal{Q}}(bf, t) = 0$. Thus, $\mathcal{P} \neq^{\mathsf{HR}}_{\mathsf{tr}} \mathcal{Q}$.
But, $\mathcal{P} =_{\mathsf{we}} \mathcal{Q}$ because every weight the observer assigns to a and b will result in an observation which is the same for both PTPs, $\mathcal{P}$ and $\mathcal{Q}$. Note that there is no internal nondeterminism which has to be resolved. Thus, the choice of an internal scheduler does not play a role.

Example 6.2 and 6.3 imply the following result.

Proposition 6.2

$=_{\mathsf{we}}$ *is incomparable to* $=_{\mathsf{tr}}^{\mathsf{HR}}$.

Note that the two counterexamples also imply that $=_{\mathsf{we}}$ is also incomparable to $=_{\mathsf{fa}}$ and $=_{\mathsf{re}}$.
Intuitively, the reason why weighted trace equivalence does not imply trace equivalence is that the observer's choice is based on the visible part of the process history but not on the complete history. Thus, if an invisible transition is taken or if the process branches probabilistically (and instantaneously), the controller adjustments are the same for each branch. However, in the following we define trace equivalence with respect to the set of schedulers that resolve external nondeterminism based on the performed trace (and internal nondeterminism based on the complete process history). We show that this variant of trace equivalence is strictly finer than weighted trace equivalence.

Before we give the definition of the scheduler class mentioned above, a remark is in order: Assume that the visible parts of two path fragments ξ and ξ', $\xi' \neq \xi$ are equal, i.e. $trace(\xi) = trace(\xi') = \sigma$ and assume that HR-scheduler $\mathcal{D}$'s resolution of external nondeterminism depends only on σ. This means that if both, $last(\xi) =: s$ and $last(\xi') =: s'$ are stable, for each pair (a, b) of visible actions the ratio of the scheduler's choice is equal i.e.

$$s \xrightarrow{a}\!\!\!\!\rightarrow, s' \xrightarrow{a}\!\!\!\!\rightarrow \text{ and } s \xrightarrow{b}\!\!\!\!\rightarrow, s' \xrightarrow{b}\!\!\!\!\rightarrow \implies \frac{\sum_\mu \mathcal{D}(\xi)(a,\mu)}{\sum_\mu \mathcal{D}(\xi)(b,\mu)} = \frac{\sum_\mu \mathcal{D}(\xi')(a,\mu)}{\sum_\mu \mathcal{D}(\xi')(b,\mu)}.$$

It is important to keep in mind that after a certain trace has been performed, different states can be reached from which visible transitions emerge having different action labels. Consider, for instance, PTP $\mathcal{Q}$ in Figure 6.5 on page 132. The two states that can be reached from the initial one with an invisible transition (trace $\sigma = \epsilon$) provide either b or c. Thus, in this case we do not stipulate any additional conditions for a trace dependent scheduler. Similarly, if $s \xrightarrow{a}\!\!\!\!\rightarrow$ and $s' \xrightarrow{a}\!\!\!\!\rightarrow$ we want the scheduler's choice to fulfill

$$\frac{\mathcal{D}(\xi)^\perp}{\sum_\mu \mathcal{D}(\xi)(a,\mu)} = \frac{\mathcal{D}(\xi')^\perp}{\sum_\mu \mathcal{D}(\xi')(a,\mu)}.$$

However, if s or s' is unstable, the situation is more complicated. Even if the environment decides for, say, action a with probability $p > 0$, $\mathcal{D}$ can choose an internal transition instead of a a-transition. For instance, in Example 6.1 on page 122 we have that in state s' action d is externally enabled with probability $\frac{2}{5}$ but internally chosen with probability $\frac{1}{4}$. Thus, a d-transition is taken with probability $\frac{2}{5} \cdot \frac{1}{4} = \frac{1}{10}$. As opposed to that, in a stable state with, say, two outgoing transitions labeled by c and d, respectively, the probability of d is $\frac{2}{5}$.
Let $\mathcal{D}(\xi, a) := \sum_{\mu} \mathcal{D}(\xi)(a, \mu)$. The above considerations motivate the following definition.

Definition 6.5 (Trace Dependent Scheduler)
We call $\mathcal{D} \in \mathsf{HR}(\mathcal{P})$ a *trace dependent scheduler* if for all $\sigma \in \mathsf{Act}^*$ there exist values $z_\perp, z_a, z_b, \ldots \in \mathbb{R}_{\geq 0}$ such that for each $\xi \in \mathsf{pathf}(\mathcal{P}, \sigma)$ with normalization constant

$$N := z_\perp + \sum_{\substack{a \in \mathsf{Act} \\ last(\xi) \xrightarrow{a}\!\!\!\!\rightarrow}} z_a$$

and $last(\xi) =: s$ the following conditions are true:

1. If s is stable then

$$\mathcal{D}(\xi)^\perp = \begin{cases} \frac{z_\perp}{N} & \text{if } N > 0, \\ 1 & \text{otherwise,} \end{cases}$$

   and $s \xrightarrow{a}\!\!\!\!\rightarrow$ implies

$$\mathcal{D}(\xi, a) = \begin{cases} \frac{z_a}{N} & \text{if } N > 0, \\ 0 & \text{otherwise.} \end{cases}$$

2. If s is unstable then $N > 0$ implies that for each $a \in \mathsf{Act}$ with $s \xrightarrow{a}\!\!\!\!\rightarrow$ there exists $p_a^\xi \in [0,1]$ with

$$\begin{aligned} \mathcal{D}(\xi, a) &= \frac{z_a}{N} \cdot p_a^\xi, \\ \mathcal{D}(\xi, \tau) &= \frac{z_\perp}{N} + \sum_{\substack{a \in \mathsf{Act} \\ s \xrightarrow{a}\!\!\!\!\rightarrow}} \frac{z_a}{N} \cdot (1 - p_a^\xi). \end{aligned}$$

If $N = 0$ then $\mathcal{D}(\xi, \tau) = 1$.

We write $\mathsf{TR} \subset \mathsf{HR}$ for the set of all trace dependent HR-schedulers[2].

Note that for each set $\mathsf{pathf}(\mathcal{P}, \sigma)$ there always exists $\xi' \in \mathsf{pathf}(\mathcal{P}, \sigma)$ such that $last(\xi')$ is stable because $\mathcal{P}$ is divergence free.

Intuitively, z_a describes the external weight of action a and $z_\perp$ corresponds to the weight of external refusal of any interaction. Similarly, p_a^ξ is the probability of a (instead of τ) being chosen internally under the condition that the external stimulus a is provided. With probability $1 - p_a^\xi$ the scheduler decides for τ.

The class TR induces the trace equivalence $=_{\mathsf{tr}}^{\mathsf{TR}}$ by setting $D = \mathsf{TR}$ in Definition 5.1 and 5.2 (see page 87).
The values z_a in the above definition (which depend on $\mathcal{D}$ and σ) correspond to the controller settings $\mathsf{ctr}_\sigma(a)$ in the weighted trace testing scenario. But the equivalence $=_{\mathsf{tr}}^{\mathsf{TR}}$ differs from weighted trace equivalence not only in that it is formulated using HR-schedulers instead of a button pushing experiment. Here, if $\mathcal{P}_1 =_{\mathsf{tr}}^{\mathsf{TR}} \mathcal{P}_2$, a scheduler $\mathcal{D}_1$ for $\mathcal{P}_1$ might use a *different* resolution of external nondeterminism than scheduler $\mathcal{D}_2$ for $\mathcal{P}_2$ to match the observations resulting from $\mathcal{P}_2$ under scheduler $\mathcal{D}_2$.

Proposition 6.3

Let $\mathcal{P}_1$ and $\mathcal{P}_2$ be PTP*s. $\mathcal{P}_1 =_{\mathsf{we}} \mathcal{P}_2$ implies $\mathcal{P}_1 =_{\mathsf{tr}}^{\mathsf{TR}} \mathcal{P}_2$. But the opposite direction does not hold.*

Sketch of Proof. Let us start with the implication. We give only a proof sketch and refer to Section A.1 of the appendix for the proof details.
Assume that $\mathcal{P}_1 =_{\mathsf{we}} \mathcal{P}_2$. Now, let $\mathcal{D}_1$ be a trace dependent scheduler for $\mathcal{P}_1$ and let $\sigma \in \mathsf{Act}^*$. According to Definition 6.5 there exist values $z_\perp, z_a, z_b, \ldots \in \mathbb{R}_{\geq 0}$ such that the above conditions are fulfilled. We perform a button pushing experiment using the weighted trace machine and assume

[2] Here, TR stands for "trace dependent, randomized".

controller settings $\mathsf{ctr}_\sigma(a) := z_a = z_a^{\mathcal{D}_1,\sigma}$ for $a \in \mathsf{Act}$, $\mathsf{ctr}_\sigma(\perp) := z_\perp = z_\sigma^{\mathcal{D}_1}(\perp)$. The idea of the proof is to construct a trace dependent scheduler $\mathcal{D}_2$ for $\mathcal{P}_2$ such that for all $t \geq 0$ and all $\sigma \in \mathsf{Act}^*$

$$\mathsf{tr}_{\mathcal{P}_1}^{\mathcal{D}_1}(\sigma, t) = \mathsf{tr}_{\mathcal{P}_2}^{\mathcal{D}_2}(\sigma, t). \tag{6.3}$$

This is done in two steps. First we construct an internal scheduler $\mathcal{E}_1$ for $\mathcal{P}_1$ based on the decisions of $\mathcal{D}_1$. Since $\mathcal{P}_1 =_{\mathsf{we}} \mathcal{P}_2$ there exists a matching internal scheduler $\mathcal{E}_2$ for $\mathcal{P}_2$. Based on $\mathcal{E}_2$ we construct $\mathcal{D}_2$ and show that Equation 6.3 holds.

For the opposite direction we consider Figure 6.5. It holds that $\mathcal{P}$ and $\mathcal{Q}$ are trace equivalent with respect to trace dependent schedulers. To see this assume that the (trace dependent) scheduler's choice for $\mathcal{P}$ is p_τ for the invisible transition, p_b and p_c for the b-transition and c-transition, respectively. A (trace dependent) scheduler for $\mathcal{Q}$ can match this by choosing the following discrete distribution in the initial state: probability $p_b/(p_b + p_c)$ is assigned to the left τ-transition and $p_c/(p_b + p_c)$ to the right τ-transition. This choice concerns internal nondeterminism. Then, in both successor states, no transition at all is chosen with probability p_τ (which is a trace dependent choice) and the respective visible transition with probability $1 - p_\tau$. Since

$$\begin{array}{lcl}
\frac{p_b}{p_b+p_c} \cdot p_\tau + \frac{p_c}{p_b+p_c} \cdot p_\tau & = & p_\tau \\
\frac{p_b}{p_b+p_c} \cdot (1 - p_\tau) = \frac{p_b}{p_b+p_c} \cdot (p_b + p_c) & = & p_b \\
\frac{p_c}{p_b+p_c} \cdot (1 - p_\tau) = \frac{p_c}{p_b+p_c} \cdot (p_b + p_c) & = & p_c
\end{array}$$

each trace has the same probability and $\mathcal{P} =_{\mathsf{tr}}^{\mathsf{TR}} \mathcal{Q}$ follows.
On the other hand $\mathcal{P} \neq_{\mathsf{we}} \mathcal{Q}$ because if the controller settings are given by, say, $\mathsf{ctr}_\epsilon(\perp) = 1$, $\mathsf{ctr}_\epsilon(b) = \mathsf{ctr}_\epsilon(c) = 2$, we get trace c with probability $3/4 \cdot 2/3 = 1/2$ if the right transition of the initial state of $\mathcal{Q}$ is (internally) chosen with probability $3/4$. But in $\mathcal{P}$ the probability of c cannot be greater than $2/5$ under these controller settings. □

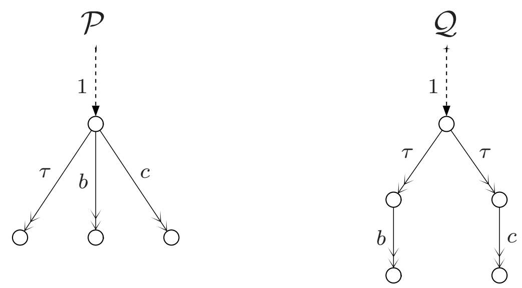

Figure 6.5: $\mathcal{P} =^{\mathsf{TR}}_{\mathsf{tr}} \mathcal{Q}$, but $\mathcal{P} \neq_{\mathsf{we}} \mathcal{Q}$

Let us now examine an example which proves that $=_{\mathsf{we}}$ fails to be a congruence with respect to the parallel composition operator of Definition 4.1 for SPTPs. Intuitively, $=_{\mathsf{we}}$ does not have the congruence property because it is not sensitive to behavior which arises from a delayed supply of external stimuli, since $=_{\mathsf{we}}$ tests for immediate interaction (or complete external refusal of an action) only.

Example 6.4

Consider the two SPTPs $\mathcal{P}_1$ and $\mathcal{P}_2$ in Figure 6.6, left. Obviously, $\mathcal{P}_1 =_{\mathsf{we}} \mathcal{P}_2$ (and also $\mathcal{P}_1 =^{\mathsf{HR}}_{\mathsf{tr}} \mathcal{P}_2$). The parallel composition of $\mathcal{P}_1$ and $\mathcal{Q}$ over $A = \{a\}$ results in a SPTP which can perform trace ac (with probability 0.5) after a delay that is PH distributed according to $(\boldsymbol{\alpha}, T)$ (compare Figure 6.6, right): the maximum of the two exponential delays (rates -1 and -3) is convoluted with the final exponential delay preceding the c-transition (rate -2). In the case of $\mathcal{P}_2 \,||_A\, \mathcal{Q}$ the delay for ac has representation $(\boldsymbol{\alpha}, T')$. This can be seen by observing that this delay is the convolution of the random variables X with distribution $F_X = F_{1,-2} \cdot F_{1,-3}$ and Y with distribution $F_{1,-1}$. From $F_{(\boldsymbol{\alpha},T)} \neq F_{(\boldsymbol{\alpha},T')}$ we derive that for all combinations of internal schedulers $\mathcal{E}_1$ and $\mathcal{E}_2$ there exists $t \geq 0$ such that

$$\mathsf{we}^{(\mathsf{ctr},\mathcal{E}_1)}_{\mathcal{P}_1 \,||_A\, \mathcal{Q}}(ac, t) \neq \mathsf{we}^{(\mathsf{ctr},\mathcal{E}_2)}_{\mathcal{P}_2 \,||_A\, \mathcal{Q}}(ac, t)$$

for all functions ctr that assign positive weights to a (initially) and c (after trace a). Thus, $\mathcal{P}_1 \,||_A\, \mathcal{Q} \neq_{\mathsf{we}} \mathcal{P}_2 \,||_A\, \mathcal{Q}$.

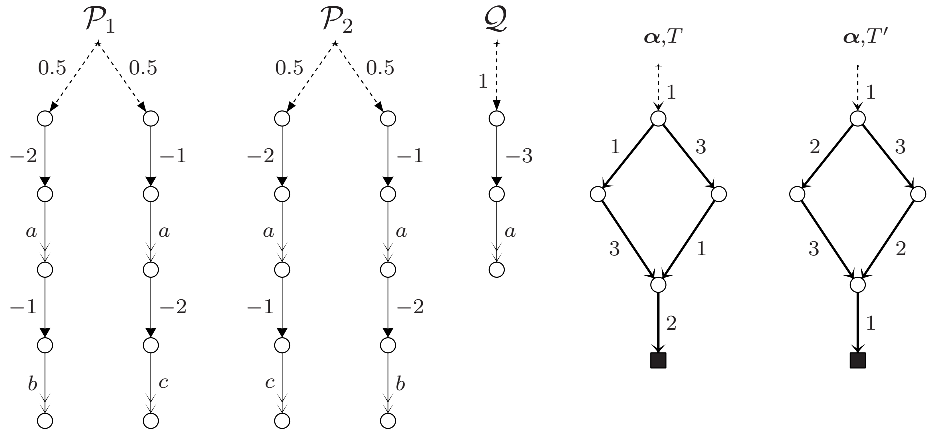

Figure 6.6: $(\mathcal{P}_1, \mathcal{P}_2) \in =_{\mathsf{we}}, =^{\mathsf{HR}}_{\mathsf{tr}}$, but $(\mathcal{P}_1 \,||_{\{a\}}\, \mathcal{Q}, \mathcal{P}_2 \,||_{\{a\}}\, \mathcal{Q}) \notin =_{\mathsf{we}}, =^{\mathsf{HR}}_{\mathsf{tr}}$ and $(\mathcal{P}_1, \mathcal{P}_2) \notin =_{\mathsf{te}}, =_{\mathsf{de}}$.

It is important to note that $=_{\mathsf{we}}$ fails to be a congruence also in the purely interleaving case of parallel composition (for a counterexample, suppose in the above example that $A = \emptyset$).

Let us now focus on the relationship between $=_{\mathsf{we}}$ and phase type bisimulation. Assume $\mathcal{P} =_{\mathsf{bs}} \mathcal{Q}$ and R is a phase type bisimulation which relates $\mathcal{P}$ and $\mathcal{Q}$. In analogy to Theorem 3.1 (compare page 61) it can be shown that for all timer settings ctr and all internal schedulers $\mathcal{E}$ there exists $\mathcal{E}'$ such that for all $k \geq 1$, $\Xi_\eta \in \mathcal{H}^R$, $|\eta| = k$

$$\mathsf{Pr}^{(\mathsf{ctr},\mathcal{E})}_{\mathcal{P}}(\{\pi \in \mathsf{path}(\mathcal{P}) \mid \pi{\downarrow_k} \in \Xi_\eta\}) = \mathsf{Pr}^{(\mathsf{ctr},\mathcal{E}')}_{\mathcal{Q}}(\{\pi \in \mathsf{path}(\mathcal{Q}) \mid \pi{\downarrow_k} \in \Xi_\eta\}).$$

But since the set of paths measured by $\mathsf{we}^{(\mathsf{ctr},\mathcal{E})}_{\mathcal{P}}(\sigma, t)$ is the (disjoint) union of sets $\Xi_\eta \in \mathcal{H}^R$, the inclusion of the following proposition follows:

Proposition 6.4

$=_{\mathsf{bs}}$ *is strictly finer than* $=_{\mathsf{we}}$.

Proof. We already justified above that the inclusion holds. Strictness can be seen by considering the two PTPs $\mathcal{P}_1$ and $\mathcal{P}_2$ in Figure 6.6. As already shown

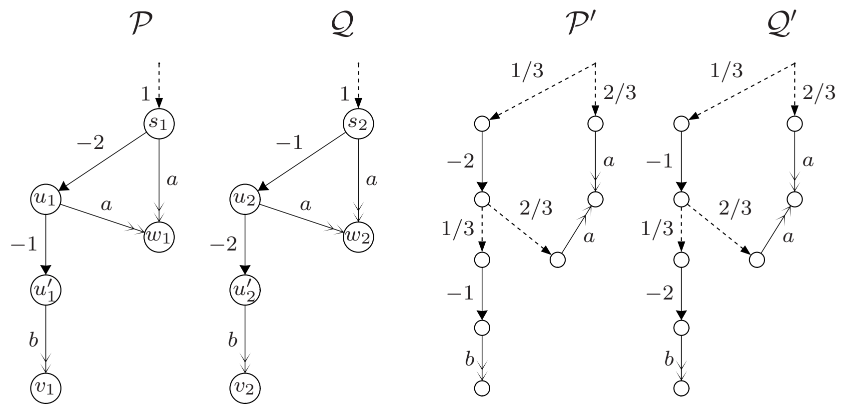

Figure 6.7: $\mathcal{P} =_{\mathsf{de}} \mathcal{Q}$ and $\mathcal{P} =_{\mathsf{te}} \mathcal{Q}$, but $\mathcal{P} \neq_{\mathsf{we}} \mathcal{Q}, \mathcal{P} \neq_{\mathsf{tr}}^{\mathsf{HR}} \mathcal{Q}$

above, $\mathcal{P}_1 =_{\mathsf{we}} \mathcal{P}_2$. But $\mathcal{P} \neq_{\mathsf{bs}} \mathcal{Q}$ because there is no phase type bisimulation that relates the target states of the a-transitions. Thus, the initial states are not related and $\nu_{\mathcal{P}_1} \not\equiv_R \nu_{\mathcal{P}_2}$ for all phase type bisimulations R. □

Although $=_{\mathsf{we}}$ is defined as a probabilistic extension of failure trace and has a close relationship to other linear-time relations such as the testing equivalence defined in [CSZ92], there are processes $=_{\mathsf{we}}$ distinguishes which one wants to treat as equal for most applications. The following example illustrates the unfavorable effect a purely time-abstract testing environment has on the distinguishing power of the resulting equivalence.

Example 6.5

Intuitively, the two SPTPs $\mathcal{P}$ and $\mathcal{Q}$ in Figure 6.7, left, show the same behavior. Both PTPs behave as follows: action a is offered and if a is not taken within a PH distributed delay with representation

$$\boldsymbol{\alpha} = \begin{bmatrix} 1 & 0 \end{bmatrix}, T = \begin{bmatrix} -1 & 1 \\ 0 & -2 \end{bmatrix},$$

action b is enabled, i.e. we can abstract from the order of the two successive exponential delays. Formally, if we combine the two successive exponential delays we see that $\mathcal{P} = \mathsf{ex}(\hat{\mathcal{P}})$ ($\mathcal{Q} = \mathsf{ex}(\hat{\mathcal{Q}})$) where $\hat{\mathcal{P}}$ ($\hat{\mathcal{Q}}$) is a copy of $\mathcal{P}$ ($\mathcal{Q}$)

but the outgoing transitions of state u_1 (u_2) are replaced by a PH transition $s_1 \xrightarrow{\boldsymbol{\alpha},T} u_1'$ ($s_2 \xrightarrow{\boldsymbol{\alpha},U} u_2'$, respectively) where

$$U = \begin{bmatrix} -2 & 2 \\ 0 & -1 \end{bmatrix}.$$

If $\mathcal{P}$ and $\mathcal{Q}$ are analyzed with respect to weighted traces, they show different behavior. Assume that $\mathsf{ctr}_\epsilon(a) = 2$, $\mathsf{ctr}_\epsilon(\perp) = 1$ and $\mathsf{ctr}_\epsilon(b) = 1$. The two SPTPs $\mathcal{P}'$ and $\mathcal{Q}'$ in Figure 6.7, right, illustrate the corresponding observations of $\mathcal{P}$ and $\mathcal{Q}$. The initial states s_1 and s_2 are replaced by two initial states, respectively. With probability 2/3 action a is executed and with probability 1/3 the PH-transition is performed. For all internal schedulers $\mathcal{E}_\mathcal{P}$, $\mathcal{E}_\mathcal{Q}$ and all $0 < t < \infty$ we get

$$\begin{aligned} \mathsf{we}_\mathcal{P}^{(\mathsf{ctr},\mathcal{E}_\mathcal{P})}(a,t) &= 2/3 + 1/3 \cdot 2/3 \cdot F_{-2}(t) \\ &\neq 2/3 + 1/3 \cdot 2/3 \cdot F_{-1}(t) = \mathsf{we}_\mathcal{Q}^{(\mathsf{ctr},\mathcal{E}_\mathcal{Q})}(a,t). \end{aligned}$$

Furthermore, $\mathcal{P}$ and $\mathcal{Q}$ are not HR-trace equivalent. This can be seen by considering a scheduler $\mathcal{D} \in \mathsf{HR}(\mathcal{P})$ which decides against the initial a transition but chooses a after the first delay. There is no matching scheduler for $\mathcal{Q}$. The problem is that $\mathcal{D}$ (and also ctr as defined above) simulates environment conditions that are in some sense not "natural" for a process acting in continuous time. An external stimulus for $\mathcal{P}$ is either already existing (when $\mathcal{P}$ enters its initial state) or provided after a certain delay. However, this is in contrast to the scenario $\mathcal{D}$ and ctr describe.

Figure 6.8 illustrates the behavior of $\mathcal{P}$ and $\mathcal{Q}$ in case that action a is externally enabled after two successive exponential delays with rate -3 and -4, respectively (and b is enabled immediately). The probability to observe a within $t \geq 0$ time units is equal for $\mathcal{P}''$ and $\mathcal{Q}''$ (this is also true for trace b). It is easily seen that $\mathcal{P}$ and $\mathcal{Q}$ exhibit equal behavior if external stimuli are provided after a certain delay (including the zero delay). In the next section we will define an equivalence that identifies $\mathcal{P}$ and $\mathcal{Q}$ because it is based on the assumption that the environment is "timed", i.e. the environment gives action inputs after a certain delay.

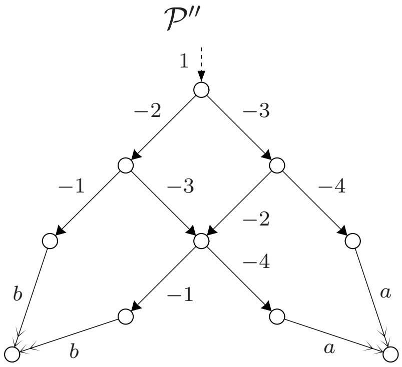

Figure 6.8: $\mathcal{P}''$ and $\mathcal{Q}''$ show equal observable behavior.

6.3 Delayed Trace Equivalence

The idea of this section is to enrich the stochastic trace machine with *countdown timers*, one for each action. The experimenter can set the timers in order to determine after which time duration the corresponding action is supplied by the environment. More precisely, an action, say, b occurs earlier as an environmental stimulus than action a if the timer of b is set to a lower value than the timer that corresponds to a (compare Figure 6.9). Therefore, the process under study might have to wait until one of the current state's transitions becomes enabled. Assume, for instance, that the current state of a PTP has only two outgoing transitions, one labeled by a, and the other one labeled by b. If now the countdown timer of b expires first, the b-transition is taken.

Each countdown timer determines the PH distribution of the random time until the corresponding action input is given. Alternatively, the observer can decide that an action is immediately available (by setting the timer to zero).

Recall that $\mathcal{R}$ is the set of all irreducible PH representations $(\boldsymbol{\alpha}, T)$ with $\boldsymbol{\alpha} \cdot \mathbf{1} = 1$. We describe the action timers by a function

$$\mathsf{tm} : \underbrace{\mathsf{Act}^*}_{\text{interaction history}} \times \underbrace{\mathsf{Act}}_{\text{timer label}} \rightarrow \underbrace{\big(\mathcal{R} \cup \{\mathsf{I}, \mathsf{B}\}\big)}_{\text{delay distribution}}$$

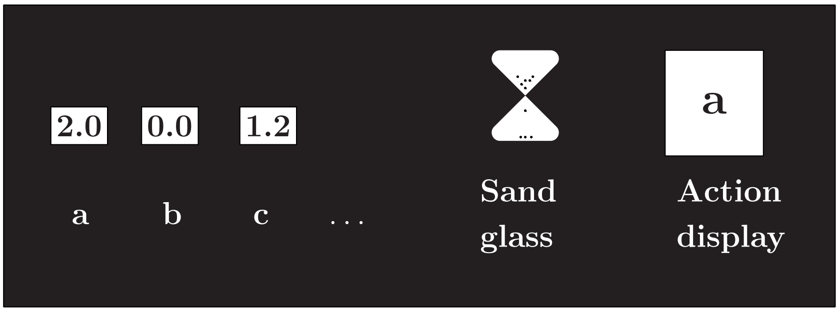

Figure 6.9: The delayed trace machine with action countdown timers

where I represents instantaneous availability of an action and B corresponds to the case where an action is completely blocked. Function $\mathsf{tm}(\tilde{\sigma}, \cdot) =: \mathsf{tm}_{\tilde{\sigma}}$ describes the timer settings after trace $\tilde{\sigma}$ has been performed. If $\mathsf{tm}_{\tilde{\sigma}}(a) = (\boldsymbol{\alpha}, T) \in \mathcal{R}$ action a is externally available after a PH distributed delay with representation $(\boldsymbol{\alpha}, T)$. We require the set

$$\{\mathsf{tm}_{\tilde{\sigma}}(a) \in \mathcal{R} \cup \{\mathsf{I}\} \mid a \in \mathsf{Act}\}$$

to be finite for all $\tilde{\sigma} \in \mathsf{Act}^*$ to avoid technical problems. This is a reasonable restriction since it means that the process acts in an environment in which only a finite number of processes want to communicate.

It is important to note that the PH delays for the external availability of an action start at that moment the last action of trace $\tilde{\sigma}$ is executed. However, if the next visible action, say, a is performed external stimuli are given according to $\mathsf{tm}_{\tilde{\sigma}a}$. External nondeterminism occurs only if several actions are immediately available, e.g. $\mathsf{tm}_{\tilde{\sigma}}(a) = \mathsf{tm}_{\tilde{\sigma}}(b) = \mathsf{I}$ and the current state of the process under study has outgoing transitions labeled by a and b, respectively.

The delayed trace scenario requires a different definition of paths because we have to record the current phase of all representations $\mathsf{tm}_{\tilde{\sigma}}(a) \in \mathcal{R}, a \in \mathsf{Act}$ while

- the process traverses through several states (via successive PH or τ-transitions) or
- other timers change their current phase, for example, if $\mathsf{tm}_{\tilde{\sigma}}(b) \in \mathcal{R}$ we ought to know which action is first externally available (a or b).

We call the execution sequences which emerge in the delayed trace scenario tm-*paths*. Let $\mathcal{G}$ be the set of functions $\mathsf{ph} : \mathsf{Act} \to \{0, 1, 2, \ldots\} \cup \{\infty\}$. which record the current phase of each action timer. For $k > 0$, we set $\mathsf{ph}(a) = k$ if representation $\mathsf{tm}_{\tilde{\sigma}}(a) = (\boldsymbol{\alpha}, T)$ is in the k-th phase (i.e. the a-timer has not yet expired but resides in state k of the Markov chain that corresponds to $(\boldsymbol{\alpha}, T)$). In the case that $\mathsf{ph}(a) = 0$, the timer of action a has expired (i.e. the absorbing state is reached). Clearly, if an action is instantaneously available, $\mathsf{ph}(a) = 0$. However, if a is blocked because $\mathsf{tm}_{\tilde{\sigma}}(a) = \mathsf{B}$ we set $\mathsf{ph}(a) = \infty$.

Definition 6.6 (tm-paths and tm-path fragments)
Let $s_1, s_2, \ldots$ be states of PTP $\mathcal{P}$, $\mathsf{e}_1, \mathsf{e}_2, \ldots \in \big(\mathbb{R}_{>0} \cup (\mathsf{Act}_\tau \times \mathbb{R}_{\geq 0})\big)$ and $\mathsf{ph}_1, \mathsf{ph}_2, \ldots \in \mathcal{G}$. An infinite or finite sequence

$$\pi = (s_1, \mathsf{ph}_1) \xrightarrow{\mathsf{e}_1} (s_2, \mathsf{ph}_2) \xrightarrow{\mathsf{e}_2} \ldots \text{ or}$$
$$\pi = (s_1, \mathsf{ph}_1) \xrightarrow{\mathsf{e}_1} (s_2, \mathsf{ph}_2) \xrightarrow{\mathsf{e}_2} \ldots \xrightarrow{\mathsf{e}_{k-1}} (s_k, \mathsf{ph}_k)$$

is a tm-*path* of $\mathcal{P}$ if the following conditions are true:

1. For $i \in \{1, 2, \ldots\}$ ($i < k$ if π is finite) let $\hat{\mathsf{e}}_i = t$ if $\mathsf{e}_i = t > 0$ and $\hat{\mathsf{e}}_i = a$ if $\mathsf{e}_i = (a, t)$. The projection $s_1 \xrightarrow{\hat{\mathsf{e}}_1} s_2 \xrightarrow{\hat{\mathsf{e}}_2} \ldots$ is an infinite or finite path of $\mathcal{P}$, respectively.
2. For all $i \in \{1, 2, \ldots\}$ ($i < k$ if π is finite) with $trace(\pi{\downarrow_i}) = \sigma$:
   (a) for all $a \in \mathsf{Act}$:

$$\mathsf{tm}_\sigma(a) \begin{cases} \in \mathcal{R} \text{ is of order } j & \text{implies} \quad 0 \leq \mathsf{ph}_i(a) \leq j, \\ = \mathsf{I} & \text{implies} \quad \mathsf{ph}_i(a) = 0, \\ = \mathsf{B} & \text{implies} \quad \mathsf{ph}_i(a) = \infty. \end{cases}$$

(b) $\mathsf{e}_i \in \mathbb{R}_{>0}$ or $\mathsf{e}_i = (a,t), t > 0$ implies $\not\exists a \in \mathsf{Act}$ with $\mathsf{ph}_i(a) = 0$, $s_i \xrightarrow{a}\!\!\!\!\rightarrow$,

(c) $\mathsf{e}_i = (a,0)$ and $a \neq \tau$ implies $\mathsf{ph}_i(a) = 0$,

(d) $\mathsf{e}_i = (a,t), t > 0$ implies $a \neq \tau$, $\mathsf{tm}_\sigma(a) \in \mathcal{R}$ and $\mathsf{ph}_i(a) > 0$.

As for paths, we require that, if π is finite, $s_k \not\longrightarrow$, $s_k \overset{\tau}{\not\twoheadrightarrow}$ and there exists no action a such that $s_k \xrightarrow{a}\!\!\!\!\rightarrow$ and $\mathsf{ph}_k(a) < \infty$. A tm-*path fragment* is a prefix of a tm-path. Let $\mathsf{path}(\mathcal{P}, \mathsf{tm})$ and $\mathsf{pathf}(\mathcal{P}, \mathsf{tm})$ denote the set of tm-paths and tm-path fragments of $\mathcal{P}$, respectively.

Informally, the first condition of Definition 6.6 states that if we remove the information about tm and all delays caused by the timers, the result is a path of $\mathcal{P}$. Condition 2a) ensures that the sequence $\mathsf{ph}_1, \mathsf{ph}_2, \ldots$ conforms to the timer settings tm. Enabled action transitions are always taken immediately (case 2b), i.e. time can only pass if immediate interaction is not possible. Condition 2c) makes sure that if a visible action is taken immediately, its action timer must have been already expired. Finally (case d), if an action transition is taken after a certain delay, it must be a visible transition (since invisible action transitions are instantaneous) and the corresponding timer imposed that delay.

For tm-paths and tm-path fragments we use similar notations as for paths. For instance, $untime(\xi)$ is a copy of ξ where all timing information is dropped.

Example 6.6

Assume that the outgoing transitions of the initial state s of PTP $\mathcal{P}$ are given by Figure 6.2 on page 122. Furthermore, let $\mathsf{tm}_\epsilon(a) = (\boldsymbol{\beta}, U)$ and $\mathsf{tm}_\epsilon(b) = (\boldsymbol{\gamma}, V)$. All remaining actions are initially blocked. Then

$$\begin{aligned}
\xi_1 &= (s, \mathsf{ph}_1) \xrightarrow{a,2.5} (v_2, \mathsf{ph}_1'), \\
\xi_2 &= (s, \mathsf{ph}_2) \xrightarrow{1.6} (v_1, \mathsf{ph}_2'), \\
\xi_3 &= (s, \mathsf{ph}_3) \xrightarrow{b,2.9} (v_4, \mathsf{ph}_3')
\end{aligned}$$

are tm-path fragments where for $i \in \{1,2,3\}$ ph_i corresponds to tm_ϵ and ph'_1 to tm_a, ph'_2 to tm_ϵ and ph'_3 to tm_b. For example, $\mathsf{ph}_2(a) = 1$ and $\mathsf{ph}'_2(a) = 2$ and $(\boldsymbol{\beta}, U)$ is of order 2.

If now $\mathsf{tm}_\epsilon(a) = \mathsf{tm}_\epsilon(b) = \mathsf{I}$ there are only three possible path fragments starting in s:

$$\begin{aligned}
\xi_1 &= (s, \mathsf{ph}_1) \xrightarrow{a,0} (v_2, \mathsf{ph}'_1) \\
\xi_2 &= (s, \mathsf{ph}_2) \xrightarrow{a,0} (v_3, \mathsf{ph}'_2) \\
\xi_3 &= (s, \mathsf{ph}_3) \xrightarrow{b,0} (v_4, \mathsf{ph}'_3)
\end{aligned}$$

Note that in the latter case the delayed trace machine which is simulating the process might encounter external nondeterminism.

Obviously, we need to define a randomized scheduler which resolves the remaining nondeterminism based on time-abstract part of the tm-path fragment executed so far[3]. In the above example, we have, for instance, nondeterminism between the two a-transitions of s and, in case that $\mathsf{tm}_\epsilon(a) = \mathsf{tm}_\epsilon(b) = \mathsf{I}$, between all transitions labeled by a or b . Furthermore, if s is the current state and $\mathsf{tm}_\epsilon(a), \mathsf{tm}_\epsilon(b) \in \mathcal{R}$ we do not know if the timer of action a expires before that one of, say, action b or vice versa. Thus, the scheduler must specify discrete distributions for both, the a-transitions and the b-transitions. In the case that immediate interaction is possible (or if s is unstable), a distribution on the successors that can be reached instantaneously is needed.

We call the pair (s, ph) *stable* if s is stable and if there exists no $a \in \mathsf{Act}$ such that $\mathsf{ph}(a) = 0$ and $s \xrightarrow{a}\!\!\!\!\rightarrow$. Furthermore, let

$$\mathsf{Act}_\tau(\mathsf{ph}) := \{a \in \mathsf{Act} \mid \mathsf{ph}(a) = 0\} \cup \{\tau\}$$

be the set of actions that are externally enabled (with respect to ph).

Definition 6.7 (tm-scheduler)

A tm-*scheduler* for PTP $\mathcal{P}$ is a function

[3]Recall that we restrict to time abstract schedulers in this chapter.

$$\mathcal{F} : \mathsf{pathf}(\mathcal{P}, \mathsf{tm}) \to \big((\mathsf{Act}_\tau \times \mathsf{dis}_S) \to [0,1]\big)$$

such that for $\xi \in \mathsf{pathf}(\mathcal{P}, \mathsf{tm})$ with $last(\xi) = (s, \mathsf{ph})$ the following conditions are true:

1. $\mathcal{F}(\xi)(a, \mu) > 0$ implies $s \xrightarrow{a}\!\!\!\!\rightarrow \mu$,
2. whenever (s, ph) is stable then $\mathcal{F}(\xi)$ is such that $s \xrightarrow{a}\!\!\!\!\rightarrow$ implies
$$\sum_{\mu : s \xrightarrow{a}\!\!\!\!\rightarrow \mu} \mathcal{F}(\xi)(a, \mu) = 1, \text{ i.e. } \mathcal{F}(\xi)(a, \cdot) \in \mathsf{dis}(\mathsf{dis}_S),$$
3. whenever (s, ph) is not stable then $\mathcal{F}(\xi) \in \mathsf{dis}(\mathsf{Act}_\tau(\mathsf{ph}) \times \mathsf{dis}_S)$,
4. for all $\xi, \xi' \in \mathsf{pathf}(\mathcal{P}, \mathsf{tm})$, whenever $untime(\xi) = untime(\xi')$ then $\mathcal{F}(\xi) = \mathcal{F}(\xi')$.

Let us examine the conditions stated above: The first condition states that a non-zero probability can only be assigned to outgoing transitions of s. The second one ensures that if s cannot be left immediately (in which case the timers resolve external nondeterminism), the scheduler acts as an internal scheduler, i.e. if the timer of action, say, a expires first and $s \xrightarrow{a}\!\!\!\!\rightarrow$, $\mathcal{F}$ chooses one of the a-transitions. In the third condition immediate interaction is possible and the scheduler can either decide for an invisible transition or for a visible transition which is permitted by both, the experimenter and $\mathcal{P}$. Finally, we achieve time independence with the fourth condition.
It is important to point out that the tm-scheduler follows a *maximal progress* strategy, i.e. $\mathcal{F}$ is forced to interact if possible and cannot refuse interaction with the environment. This stands in contrast to HR-schedulers which are allowed to choose subdistributions λ with $\lambda^\perp > 0$.

A tm-path

$$\pi = (s_1, \mathsf{ph}_1) \xrightarrow{\mathsf{e}_1} (s_2, \mathsf{ph}_2) \xrightarrow{\mathsf{e}_2} \dots$$

is called a $(\mathsf{tm}, \mathcal{F})$-*path* if $\pi \in \mathsf{path}(\mathcal{P}, \mathsf{tm})$ respects the decisions of $\mathcal{F}$, i.e., for all $i \geq 1$ ($i < k$ if π is of finite length k) we require the following:

Whenever $\mathsf{e}_i = a$ or $\mathsf{e}_i = (a, t)$ then there exists $\mu_i \in \mathsf{dis}_S$ with $\mu_i(s_{i+1}) > 0$ and $\mathcal{F}(\pi\downarrow_i)(a, \mu_i) > 0$. A $(\mathsf{tm}, \mathcal{F})$-path fragment is a path fragment of a $(\mathsf{tm}, \mathcal{F})$-path.

We construct a probability measure $\mathsf{Pr}^{(\mathsf{tm},\mathcal{F})}$ in a similar way to Section 6.2 by specifying the probability of cylinder sets of $(\mathsf{tm}, \mathcal{F})$-paths. However, in the setting of delayed traces, the construction is more complicated than in the previous sections since we have to incorporate the action timers.
Let us fix $\mathcal{P}, \mathcal{F}$ and tm. Let $\mathsf{C}_{(s,\mathsf{ph})}$ be the cylinder set of all $(\mathsf{tm}, \mathcal{F})$-paths that start with (s, ph). We define

$$\mathsf{Pr}^{(\mathsf{tm},\mathcal{F})}(\mathsf{C}_{(s,\mathsf{ph})}) = \nu(s) \cdot \mathsf{Pr}(\mathsf{ph}, \epsilon)$$

where ν is the initial distribution of $\mathcal{P}$ and the probability of an action timer initialization according to ph is given by

$$\mathsf{Pr}(\mathsf{ph}, \sigma) := \begin{cases} \prod\limits_{b:\mathsf{tm}_\sigma(b)=(\boldsymbol{\alpha},T)\in\mathcal{R}} \boldsymbol{\alpha}(\mathsf{ph}(b)) & \text{if } \exists b : \mathsf{tm}_\sigma(b) \in \mathcal{R}, \\ 1 & \text{otherwise} \end{cases} \tag{6.4}$$

if the timer settings correspond to tm_σ, $\sigma \in \mathsf{Act}^*$.
Assume now, that $k > 1$ and let

$$\zeta = (s_1, \mathsf{ph}_1)\ E_1\ (s_2, \mathsf{ph}_2)\ E_2 \ldots\ E_{k-1}\ (s_k, \mathsf{ph}_k)$$

where $s_1, s_2, \ldots, s_k \in S$, $\mathsf{ph}_1, \mathsf{ph}_2, \ldots, \mathsf{ph}_k \in \mathcal{G}$ and for $i \in \{1, 2, \ldots, k-1\}$ either

- $E_i = \{(a, 0)\}$ for some $a \in \mathsf{Act}_\tau$,
- $E_i = \{a\} \times (x, y]$ for some $a \in \mathsf{Act}$, $0 \le x < y$ or
- $E_i = (x, y]$, $0 \le x < y$.

Cylinder set C_ζ is then the set of all $(\mathsf{tm}, \mathcal{F})$-paths

$$\pi = (s_1, \mathsf{ph}_1) \xrightarrow{\mathsf{e}_1} (s_2, \mathsf{ph}_2) \xrightarrow{\mathsf{e}_2} \ldots$$

with $\mathsf{e}_i \in E_i$ for $i \in \{1, 2, \ldots, k-1\}$. Furthermore, let $trace(\pi\downarrow_{k-1}) = \sigma$, $\mathcal{F}(\pi\downarrow_{k-1}) = \lambda$. and

$$\zeta' = (s_1, \mathsf{ph}_1)\ E_1\ (s_2, \mathsf{ph}_2)\ E_2 \ldots\ E_{k-1}\ (s_{k-1}, \mathsf{ph}_{k-1}).$$

We distinguish the following cases:

1. If $last(\zeta') = (s_{k-1}, \mathsf{ph}_{k-1})$ is stable a race between the outgoing PH transitions of s_{k-1} and the action timers of elements of the (finite) set[4]

$$A := \{a \in \mathsf{Act} \mid \mathsf{tm}_\sigma(a) = (\boldsymbol{\alpha}, T) \in \mathcal{R} \wedge s_{k-1} \xrightarrow{a}\!\!\!\!\rightarrow \}$$

takes place. This race is reflected by a SPTP $\mathcal{Q}$ which is the parallel composition of the following SPTPs. The first component is given by $\mathcal{P}' = \mathsf{ex}(S_\mathcal{P}, \longrightarrow_\mathcal{P}, \emptyset, \delta_{s_{k-1}})$ and the remaining components are SPTPs $\mathcal{P}_a = (S_a, \longrightarrow_a, \emptyset, \delta_{\mathsf{ph}_{k-1}(a)})$, $a \in A$, with set of states $S_a := \{0, 1, \ldots, l\}$ and[5]

$$\longrightarrow_a = \{(i, -T_{ij}, j) \mid 0 < i, j \leq l, i \neq j\} \cup \{(i, -\mathbf{T}^0_i, 0) \mid 0 < i \leq l\}$$

where $\mathsf{tm}_\sigma(a) = (\boldsymbol{\alpha}, T)$ is of order l.

   (a) Assume that E_{k-1} is an interval $(x, y]$. A PH transition was taken before a timer of one of the actions $a \in A$ expired. Let $S_\mathcal{Q}$ be the set of states of $\mathcal{Q}$ and $B \subseteq S_\mathcal{Q}$ the set of all states for which the local state $u^{(j)}$ of $\mathcal{P}'$ fulfills $u = s_{k-1}$ and for all $a \in A$ the local state u_a of $\mathcal{P}_a$ is not zero. Furthermore, $C \subseteq S_\mathcal{Q}$ is the set of all states for which the local state $v^{(j)}$ of $\mathcal{P}'$ fulfills $v = s_k$ and for all $a \in A$ the local state v_a of $\mathcal{P}_a$ is such that $v_a = \mathsf{ph}_k(a)$. The probability that the timer of a reaches phase $\mathsf{ph}_k(a)$ from phase $\mathsf{ph}_{k-1}(a)$ and the current state of $\mathcal{P}$ changes from s_{k-1} to s_k within interval $(x, y]$ is given by $\mathsf{reach}^\mathcal{Q}(B, C, (x, y])$.

   Thus, we get

$$\mathsf{Pr}^{(\mathsf{tm},\mathcal{F})}(\mathsf{C}_\zeta) \quad = \quad \mathsf{Pr}^{(\mathsf{tm},\mathcal{F})}(\mathsf{C}_{\zeta'}) \cdot \mathsf{reach}^\mathcal{Q}(B, C, (x, y]). \qquad (6.5)$$

[4]Recall that the set $\{\mathsf{tm}_{\tilde\sigma}(a) \in \mathcal{R} \mid a \in \mathsf{Act}\}$ is supposed to be finite for all $\tilde\sigma \in \mathsf{Act}^*$.
[5]Recall that $\mathbf{T^0} = -T\mathbf{1}$.

(b) If $E_{k-1} = \{a\} \times (x, y]$, $a \in \mathsf{Act}$ then the probability to perform a within interval $(x, y]$ is composed of three factors:

   i. The probability that the a-timer expires earlier than all the other timers and before a PH transition of s_{k-1} can be taken (within $(x, y]$).

   ii. The probability that after a is performed the new timer settings correspond to ph_k.

   iii. The probability that the tm-scheduler decides for a certain a-transition.

   We proceed by calculating the three factors as follows:

   i. Consider again SPTP $\mathcal{Q}$ of case 1. Let $B \subseteq S_\mathcal{Q}$ be defined as above and let $D \subseteq S_\mathcal{Q}$ be the set of all states for which the local state $v^{(j)}$ of $\mathcal{P}'$ fulfills $v = s_{k-1}$ and the local state v_a of $\mathcal{P}_a$ is such that $v_a = 0$ whereas the local states v_b of all $\mathcal{P}_b$, $a \neq b$, fulfill $v_b > 0$. Then

$$p_1 := \mathsf{reach}^\mathcal{Q}(B, D, (x, y]) \tag{6.6}$$

   is the sought-after probability.

   ii. The probability of an action timer initialization according to ph_k is defined as $p_2 := \mathsf{Pr}(\mathsf{ph}_k, \sigma a)$.

   iii. Finally, recall that $\mathcal{F}(\pi{\downarrow}_{k-1}) = \lambda$. The third factor is given by

$$p_3 := \sum \left\{\!\left| \lambda(a, \mu) \cdot \mu(s_k) \mid \exists \mu : s_{k-1} \xrightarrow{a} \mu \right|\!\right\}. \tag{6.7}$$

   Putting it all together, we get

$$\mathsf{Pr}^{(\mathsf{tm},\mathcal{F})}(\mathsf{C}_\zeta) = \mathsf{Pr}^{(\mathsf{tm},\mathcal{F})}(\mathsf{C}_{\zeta'}) \cdot p_1 \cdot p_2 \cdot p_3.$$

2. Now, let $(s_{k-1}, \mathsf{ph}_{k-1})$ be unstable. Then $E_{k-1} = \{(a, 0)\}$, $a \in \mathsf{Act}_\tau$. This implies that $s_{k-1} \xrightarrow{a}$ and $\mathsf{ph}_{k-1}(a) = 0$ if $a \neq \tau$. Let p_2 and p_3 be defined as above except that $p_2 := 1$ if $a = \tau$.

We set

$$\mathsf{Pr}^{(\mathsf{tm},\mathcal{F})}(\mathsf{C}_{\zeta}) := \mathsf{Pr}^{(\mathsf{tm},\mathcal{F})}(\mathsf{C}_{\zeta'}) \cdot p_2 \cdot p_3.$$

Measure $\mathsf{Pr}^{(\mathsf{tm},\mathcal{F})}$ can be extended to a unique probability measure for sets of $(\mathsf{tm},\mathcal{F})$-paths (similar as in Section 3.6, page 52). Sometimes, we may take as sample space the set of all tm-paths instead of the set of $(\mathsf{tm},\mathcal{F})$-paths by assuming that sets of paths that are prohibited by $\mathcal{E}$ have probability zero. Moreover, we may write $\mathsf{Pr}_{\mathcal{P}}^{(\mathsf{tm},\mathcal{F})}$ instead of $\mathsf{Pr}^{(\mathsf{tm},\mathcal{F})}$.

Definition 6.8 (Delayed Trace Observation)
Let $\mathcal{P}$ be a PTP and let the timer settings of the delayed trace machine be given by

$$\mathsf{tm} : \mathsf{Act}^* \times \mathsf{Act} \rightarrow \big(\mathcal{R} \cup \{\mathsf{I}, \mathsf{B}\}\big).$$

Furthermore, let $\mathcal{F}$ be a tm-scheduler for $\mathcal{P}$. A *delayed trace observation* is a function

$$\mathsf{de}_{\mathcal{P}}^{(\mathsf{tm},\mathcal{F})} : \big(\mathsf{Act}^* \times \mathbb{R}_{\geq 0}\big) \rightarrow [0,1]$$

such that

$$\mathsf{de}_{\mathcal{P}}^{(\mathsf{tm},\mathcal{F})}(\sigma,t) \;=\; \mathsf{Pr}_{\mathcal{P}}^{(\mathsf{tm},\mathcal{F})}(\{\pi \in \mathsf{path}(\mathcal{P},\mathsf{tm}) \mid \exists i : \mathit{trace}(\pi{\downarrow_i}) = \sigma, \; \mathit{time}(\pi{\downarrow_i}) \leq t\}).$$

The *set of delayed trace observations* of $\mathcal{P}$ with respect to tm is defined as

$$\mathsf{O}_{\mathsf{de}}^{\mathsf{tm}}(\mathcal{P}) = \{\mathsf{de}_{\mathcal{P}}^{(\mathsf{tm},\mathcal{F})} \mid \mathcal{F} \text{ is a } \mathsf{tm}\text{-scheduler for } \mathcal{P}\}.$$

Definition 6.9 (Delayed Trace Equivalence)
Two PTPs $\mathcal{P}_1$ and $\mathcal{P}_2$ are *delayed trace equivalent*, written $\mathcal{P}_1 =_{\mathsf{de}} \mathcal{P}_2$, iff for all functions $\mathsf{tm} : \mathsf{Act}^* \times \mathsf{Act} \rightarrow \big(\mathcal{R} \cup \{\mathsf{I}, \mathsf{B}\}\big)$

$$\mathsf{O}_{\mathsf{de}}^{\mathsf{tm}}(\mathcal{P}_1) = \mathsf{O}_{\mathsf{de}}^{\mathsf{tm}}(\mathcal{P}_2).$$

Next, we discuss the relationship between failure trace equivalence (compare Definition 6.4 on page 124) and delayed trace equivalence. If we allow only for timer settings that either block an action completely or enable it immediately, there is a direct obvious relationship between $=_{\mathsf{de}}$ and $=_{\mathsf{ftr}}$.

Proposition 6.5

Let $\mathcal{P}_1$ *and* $\mathcal{P}_2$ *be* PTP*s. Then* $\mathcal{P}_1 =_{\mathsf{ftr}} \mathcal{P}_2$ *iff for all functions* $\mathsf{tm} : \mathsf{Act}^* \times \mathsf{Act} \to \big(\{\mathsf{I}, \mathsf{B}\}\big)$ *and all* tm*-schedulers* $\mathcal{F}_1$ *for* $\mathcal{P}_1$ *there exists a* tm*-scheduler* $\mathcal{F}_2$ *for* $\mathcal{P}_2$ *such that*

$$\lim_{t\to\infty} \mathsf{de}_{\mathcal{P}_1}^{(\mathsf{ctr},\mathcal{F}_1)}(\sigma, t) > 0 \iff \lim_{t\to\infty} \mathsf{de}_{\mathcal{P}_2}^{(\mathsf{ctr},\mathcal{F}_2)}(\sigma, t) > 0.$$

In the sequel, we will see that the above proposition is no longer valid if $\mathsf{tm} : \mathsf{Act}^* \times \mathsf{Act} \to \big(\mathcal{R} \cup \{\mathsf{I}, \mathsf{B}\}\big)$.

Our next objective is the relationship between $=_{\mathsf{de}}$ and the relations $=_{\mathsf{tr}}^{\mathsf{HR}}$ and $=_{\mathsf{we}}$.

Proposition 6.6

Relation $=_{\mathsf{de}}$ *is incomparable to* $=_{\mathsf{tr}}^{\mathsf{HR}}$ *and* $=_{\mathsf{we}}$*.*

Proof.

Neither $=_{\mathsf{we}}$ *nor* $=_{\mathsf{tr}}^{\mathsf{HR}}$ *implies* $=_{\mathsf{de}}$: This is due to the fact that $=_{\mathsf{de}}$ is sensitive to the amount of time a visible action is offered to the process under study: Recall $\mathcal{P}_1$ and $\mathcal{P}_2$ in Example 6.4 (compare Figure 6.6 on page 133) with $\mathcal{P}_1 =_{\mathsf{we}} \mathcal{P}_2$ and $\mathcal{P}_1 =_{\mathsf{tr}}^{\mathsf{HR}} \mathcal{P}_2$ as already shown. It holds that $\mathcal{P}_1 \neq_{\mathsf{de}} \mathcal{P}_2$. To see this, assume that the timer of a expires after an exponential delay, say with rate $r = -3$. After trace a is performed, action c is immediately available. Now, consider all observations with trace ac. In the case of $\mathcal{P}_1$ the right a-transition is offered after the PH transition with rate -1 is taken while in the case of $\mathcal{P}_2$ we have rate -2 for the left a-transition. The two PH representations in Figure 6.6, right, describe the distribution of the time until trace ac is performed, respectively. The probability of observing ac is different in the two PTPs (independent of the chosen scheduler) as the representations describe different distributions.

$=_{\mathsf{de}}$ *does not imply* $=_{\mathsf{we}}$ *or* $=_{\mathsf{tr}}^{\mathsf{HR}}$: Let us consider Example 6.5 on page 134. The two PTPs $\mathcal{P}$ and $\mathcal{Q}$ in Figure 6.7 are not in $=_{\mathsf{we}}$ and not in $=_{\mathsf{tr}}^{\mathsf{HR}}$ as already shown. But $\mathcal{P} =_{\mathsf{de}} \mathcal{Q}$. To see this, let us first consider an example. The PTPs $\mathcal{P}''$ and $\mathcal{Q}''$ in Figure 6.8 on page 136 illustrate the behavior of $\mathcal{P}$ and $\mathcal{Q}$ if the timer settings are $\mathsf{tm}_\epsilon(a) = (\boldsymbol{\alpha}, T)$ where

$$\boldsymbol{\alpha} = \begin{bmatrix} 1 & 0 \end{bmatrix}, T = \begin{bmatrix} -3 & 3 \\ 0 & -4 \end{bmatrix}$$

and $\mathsf{tm}_\epsilon(b) = \mathsf{I}$. It holds that for arbitrary tm-schedulers $\mathcal{F}$ and $\mathcal{F}'$ and for all $t \geq 0$

$$\mathsf{de}_{\mathcal{P}}^{(\mathsf{tm},\mathcal{F})}(a,t) = \mathsf{de}_{\mathcal{Q}}^{(\mathsf{tm},\mathcal{F}')}(a,t) \text{ and } \mathsf{de}_{\mathcal{P}}^{(\mathsf{tm},\mathcal{F})}(b,t) = \mathsf{de}_{\mathcal{Q}}^{(\mathsf{tm},\mathcal{F}')}(b,t).$$

This comes from the fact that both, $\mathcal{P}''$ and $\mathcal{Q}''$, perform b after a delay distributed according to $F_{\boldsymbol{\alpha},T}$ and a after a delay distributed according to $F_{\boldsymbol{\alpha},U} = F_{\boldsymbol{\alpha},V}$ where

$$U = \begin{bmatrix} -1 & 1 \\ 0 & -2 \end{bmatrix}, V = \begin{bmatrix} -2 & 2 \\ 0 & -1 \end{bmatrix}.$$

Now, in general we argue as follows: Obviously, $\mathsf{tm}_\epsilon(a) \in \mathcal{R}$ is the only interesting case. Assume that F is the distribution that corresponds to $\mathsf{tm}_\epsilon(a)$. Let X be a random variable with distribution F and let Y be distributed according to $F_{\boldsymbol{\alpha},U} = F_{\boldsymbol{\alpha},V}$. For both, $\mathcal{P}$ and $\mathcal{Q}$, trace a is performed if $X < Y$ and b (if b is immediately possible) otherwise. Thus, $\mathcal{P} =_{\mathsf{de}} \mathcal{Q}$.

□

Note that $=_{\mathsf{de}}$ is also not comparable to failure or ready equivalence since we can use the same counterexamples as above for a comparison with failure or ready equivalence.

Obviously, it is possible to define a scheduler class D such that $\mathcal{P} =_{\mathsf{de}} \mathcal{Q}$ implies $\mathcal{P} =^{D}_{\mathsf{tr}} \mathcal{Q}$. More precisely, the scheduler can choose a PH distribution for each action and this choice exclusively depends on the visible part of the history, i.e. on the trace (similar as in the case of trace-dependent schedulers, compare Definition 6.5 on page 129). However, we omit the details here because a similar construction has been made in the previous section (compare Proposition 6.3).

One might expect $=_{\mathsf{de}}$ to be a congruence with respect to the parallel composition of SPTPs because the counterexample we gave for weighted trace equivalence does not work for delayed trace equivalence. A closer look on this example (see Figure 6.6 on page133) shows that the idea was to find SPTPs $\mathcal{P}_1$, $\mathcal{P}_2$ such that

- $\mathcal{P}_1$ and $\mathcal{P}_2$ cannot be distinguished by a time-abstract environment
- in the parallel composition with some SPTP $\mathcal{Q}$, a visible action is enabled after a delay which has a different distribution in $\mathcal{P}_1 \,||_A\, \mathcal{Q}$ and $\mathcal{P}_2 \,||_A\, \mathcal{Q}$.

Relation $=_{\mathsf{de}}$ can "detect" such differences by simulating an environment in which actions are enabled after a certain delay. For instance, if we choose timer settings $\mathsf{tm}_\epsilon(a) = (1, -3) \in \mathcal{R}$, $\mathsf{tm}_a(b) = \mathsf{B}$ and $\mathsf{tm}_a(c) = \mathsf{I}$, we get different distributions for the time to observe trace ac. Thus, $\mathcal{P}_1 \neq_{\mathsf{de}} \mathcal{P}_2$.

However, $=_{\mathsf{de}}$ fails to be a congruence. This is illustrated by the following counterexample.

Example 6.7

Let us first go back to Figure 6.3 on page 126. The two PTPs $\mathcal{P}$ and $\mathcal{Q}$

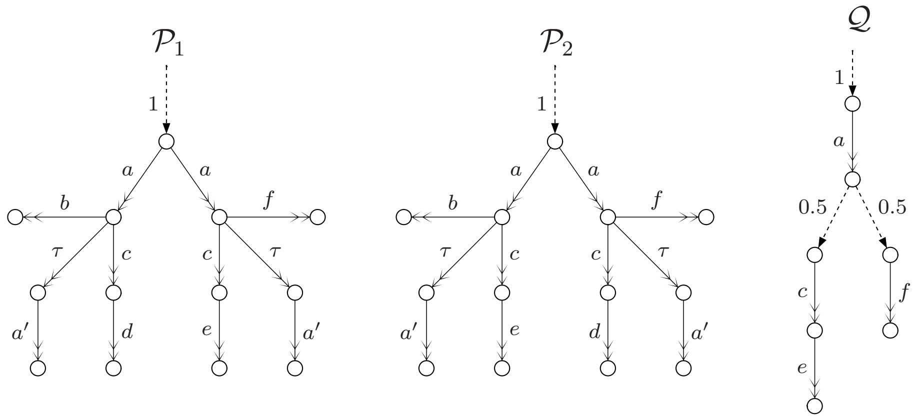

Figure 6.10: $(\mathcal{P}_1, \mathcal{P}_2) \notin =_{\mathsf{we}}, =_{\mathsf{te}}$, but $\mathcal{P}_1 =_{\mathsf{de}} \mathcal{P}_2$

can only be distinguished if external nondeterminism is resolved probabilistically, for example, if after trace a both, c and f are externally enabled with a non-zero probability (compare Example 6.2). If we now force the delayed trace machine to let this nondeterministic choice be unresolved they cannot be distinguished. This is achieved by adding τ-transitions to the respective states. The resulting processes $\mathcal{P}_1, \mathcal{P}_2$ are illustrated in Figure 6.10. They cannot be distinguished by $=_{\mathsf{de}}$. This can be seen from the following considerations: The only interesting case for tm_ϵ is $\mathsf{tm}_\epsilon(a) \in \mathcal{R} \cup \{\mathsf{I}\}$. In the case of tm_a two assignments are of interest: $\mathsf{tm}_a(c) = \mathsf{tm}_a(f) = \mathsf{I}$, $\mathsf{tm}_a(b) = \mathsf{B}$ or $\mathsf{tm}_a(c) = \mathsf{tm}_a(b) = \mathsf{I}$, $\mathsf{tm}_a(f) = \mathsf{B}$. Due to symmetry reasons, it is sufficient to consider one case, say, the former one. Then $\mathsf{tm}_{ac}(e) = \mathsf{I}$, $\mathsf{tm}_{ac}(d) = \mathsf{B}$ is the only interesting choice for tm_{ac}. It is not hard to see that with this definition of tm, each tm-scheduler for $\mathcal{P}_1$ can find a matching tm-scheduler for $\mathcal{P}_2$ and vice versa. Thus, $\mathcal{P}_1 =_{\mathsf{de}} \mathcal{P}_2$.

Now, we examine $\mathcal{P}_1 \,||_{\mathsf{Act}}\, \mathcal{Q}$ and $\mathcal{P}_2 \,||_{\mathsf{Act}}\, \mathcal{Q}$ (see Figure 6.11). Assume that all visible actions are always immediately possible. A tm-scheduler for $\mathcal{P}_1 \,||_{\mathsf{Act}}\, \mathcal{Q}$ may choose the right branch in the initial state with probability one. The subsequent choices are such that both, trace af and trace ace are observed with probability 0.5. A tm-scheduler for $\mathcal{P}_2 \,||_{\mathsf{Act}}\, \mathcal{Q}$ must choose the left branch with probability one to match the probability of trace ace. But then af has

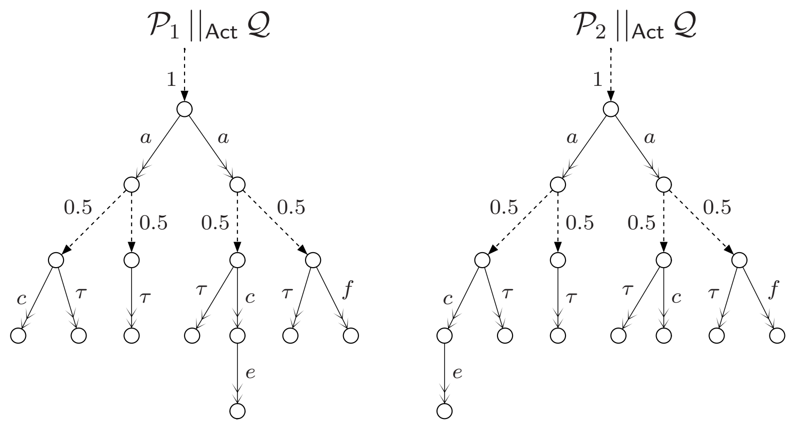

Figure 6.11: $\mathcal{P}_1 \,||_{\mathsf{Act}}\, \mathcal{Q}$ and $\mathcal{P}_2 \,||_{\mathsf{Act}}\, \mathcal{Q}$

probability zero. Therefore, $\mathcal{P}_1 \,||_{\mathsf{Act}}\, \mathcal{Q} \neq_{\mathsf{de}} \mathcal{P}_2 \,||_{\mathsf{Act}}\, \mathcal{Q}$.

The above example shows that $=_{\mathsf{de}}$ fails to detect different behaviors of unstable states and one might argue that, if several actions are externally available at a certain time instant, the nondeterminism between them should be resolved by the experimenter. However, this would contradict our intuition of two actions both having delay zero: The environment of process $\mathcal{P}$ might consist of several components interacting with $\mathcal{P}$. If two of them offer actions that $\mathcal{P}$ is able to respond to, $\mathcal{P}$ can decide which one to take. We will discuss this issue more detailed in the last section (see Remark 6.1).

We now turn to a comparison of phase type bisimulation and $=_{\mathsf{de}}$.

Proposition 6.7

$=_{\mathsf{bs}}$ *is strictly finer than* $=_{\mathsf{de}}$.

Proof. If we restrict to SPTPs the inclusion of the following proposition is a direct implication of Theorem 7.1 (on page 159) and Proposition 7.3 (on page 163). However, for the general case we have to show that if $\mathcal{P} =_{\mathsf{bs}} \mathcal{Q}$ then for all functions tm and all tm-schedulers $\mathcal{F}_1$ for $\mathcal{P}$ there exists a tm-

scheduler $\mathcal{F}_2$ for $\mathcal{Q}$ such that

$$\begin{aligned} & \mathsf{Pr}_{\mathcal{P}}^{(\mathsf{tm},\mathcal{F}_1)}(\{\pi \in \mathsf{path}(\mathcal{P},\mathsf{tm}) \mid \pi{\downarrow_k} \in \Xi_\eta\}) \\ = \ & \mathsf{Pr}_{\mathcal{Q}}^{(\mathsf{tm},\mathcal{F}_2)}(\{\pi \in \mathsf{path}(\mathcal{Q},\mathsf{tm}) \mid \pi{\downarrow_k} \in \Xi_\eta\}) \end{aligned} \tag{6.8}$$

for all sets Ξ_η of tm-path fragments. Here, η is a sequence

$$\eta = (A_1, \mathsf{ph}_1)\ E_1\ (A_2, \mathsf{ph}_2)\ E_2 \ldots E_{k-1}\ (A_k, \mathsf{ph}_k)$$

where $\mathsf{ph}_1, \mathsf{ph}_2, \ldots, \mathsf{ph}_k \in \mathcal{G}$ and for $i \in \{1, 2, \ldots, k-1\}$ either

- ⋄ $E_i = \{(a, 0)\}$ for some $a \in \mathsf{Act}_\tau$,
- ⋄ $E_i = \{a\} \times (x, y]$ for some $a \in \mathsf{Act}$, $0 \leq x < y$ or
- ⋄ $E_i = (x, y]$, $0 \leq x < y$.

and $A_1, A_2, \ldots, A_k$ are equivalence classes of a bisimulation equivalence relating $\mathcal{P}$ and $\mathcal{Q}$. The proof of Equation (6.8) is by induction on k and goes along very similar lines as that of Lemma 3.1
It remains to prove strictness. Consider $\mathcal{P}$ and $\mathcal{Q}$ in Figure 6.7 on page 134. As already shown in the proof of Proposition 6.6 we have $\mathcal{P} =_{\mathsf{de}} \mathcal{Q}$. But $\mathcal{P} \neq_{\mathsf{bs}} \mathcal{Q}$ because the successor states of s_1 and s_2 cannot be related by any phase type bisimulation R. This is because u_1 and u_2 have PH transitions with different distributions leading to equivalence class $[u_1'] = [u_2']$. □

Finally, we remark that, in general, $\mathcal{P} \neq_{\mathsf{de}} \mathsf{ex}(\mathcal{P})$. Intuitively, this comes from the fact that a tm-scheduler $\mathcal{F}$ can make different choices in states representing different phases of a state. However, we prove a similar result as stated in Theorem 4.1 (see page 76). As opposed to the situation in Theorem 4.1 we are now able to observe that in PTP $\mathcal{P}$ action transitions are taken after remaining for a certain amount of time in a state. Thus, all tm-path fragments in $\mathsf{ex}(\mathcal{P})$ are a prefix of a $\mathcal{P}$-observable path fragment (where we adapt the definitions of Section 4.3 in the obvious way). Consequently, operator contr replaces every maximal subsequence

$$(s_1^{(k_1)}, \mathsf{ph}_1) \xrightarrow{\mathsf{e}_1} (s_2^{(k_2)}, \mathsf{ph}_2) \xrightarrow{\mathsf{e}_2} \ldots \xrightarrow{\mathsf{e}_j} (s_{j+1}^{(k_{j+1})}, \mathsf{ph}_{j+1})$$

of a tm-path fragment ξ with

- $s_1 = s_i$, $k_i \in \{1, 2, \ldots, n_{s_1}\}$ for $1 \leq i \leq j$,
- $\mathsf{e}_1, \ldots, \mathsf{e}_{j-1} \in \mathbb{R}_{>0}$ if $j > 1$,
- $s_1 \neq s_{j+1}$ if $\mathsf{e}_j \in \mathbb{R}_{>0}$,
- $k_{j+1} \in \{1, 2, \ldots, n_{s_{j+1}}\}$

by

$$(s_1, \mathsf{ph}_1) \xrightarrow{\mathsf{e}} (s_{j+1}, \mathsf{ph}_{j+1})$$

where

$$\mathsf{e} = \begin{cases} \sum_{i=1}^{j} \mathsf{e}_i & \text{if } \mathsf{e}_j \in \mathbb{R}_{>0}, \\ (a, t) \text{ with } t = \sum_{i=1}^{j-1} \mathsf{e}_i + t' & \text{if } \mathsf{e}_j = (a, t') \in \mathsf{Act}_\tau \times \mathbb{R}_{\geq 0}. \end{cases}$$

It is not hard to see that for a $\mathcal{P}$-observable path fragment $\xi \in \mathsf{pathf}(\mathsf{ex}(\mathcal{P}), \mathsf{tm})$ we have $\mathsf{contr}(\xi) \in \mathsf{pathf}(\mathcal{P}, \mathsf{tm})$.

Example 6.8
Recall Example 4.2 on page 72. Let

$$(s^{(1)}, \mathsf{ph}_1) \xrightarrow{0.2} (u^{(1)}, \mathsf{ph}_2) \xrightarrow{0.3} (u^{(2)}, \mathsf{ph}_3) \xrightarrow{a,0.5} (s^{(1)}, \mathsf{ph}_4)$$

be a tm-path fragment in $\mathcal{Q} = \mathsf{ex}(\mathcal{P})$. Then

$$\mathsf{contr}(\xi) = (s, \mathsf{ph}_1) \xrightarrow{0.2} (u, \mathsf{ph}_2) \xrightarrow{a,0.8} (s, \mathsf{ph}_4) \in \mathsf{pathf}(\mathcal{P}, \mathsf{tm}).$$

In a similar way to Section 4.3 (see page 75) we define for a given cylinder set C_ζ, $|\zeta| = k$ of tm-paths in $\mathcal{P}$ that $\mathsf{ex}(\mathsf{C}_\zeta)$ is the set of all tm-paths $\pi \in \mathsf{path}(\mathsf{ex}(\mathcal{P}), \mathsf{tm})$ for which there exists $j \geq k$ such that $\pi{\downarrow_j}$ is $\mathcal{P}$-observable and $\mathsf{contr}(\pi{\downarrow_j})$ is the prefix of length k of a path in C_ζ. Let $\mathcal{C}_{\mathcal{P}}^{\mathsf{tm}}$ be the set of all such cylinder sets.

The following Theorem highlights the relationship between $=_{\mathsf{de}}$ and the ex-operator. The proof details can be found in Appendix A.2. They are similar to those of Theorem 4.1.

Theorem 6.1
Let $\mathcal{P}$ be a PTP *and* $\mathsf{tm} : \mathsf{Act}^* \times \mathsf{Act} \to \big(\mathcal{R} \cup \{\mathsf{I}, \mathsf{B}\}\big)$. *Then for each* tm-*scheduler* $\mathcal{F}'$ *for* $\mathcal{P}$ *there exists a* tm-*scheduler* $\mathcal{F}$ *for* $\mathsf{ex}(\mathcal{P})$ *such that for all* $\mathsf{C}_\zeta \in \mathcal{C}^{\mathsf{tm}}_{\mathcal{P}}$

$$\mathsf{Pr}^{(\mathsf{tm},\mathcal{F}')}_{\mathcal{P}}(\mathsf{C}_\zeta) = \mathsf{Pr}^{(\mathsf{tm},\mathcal{F})}_{\mathsf{ex}(\mathcal{P})}(\mathsf{ex}(\mathsf{C}_\zeta)).$$

Note that the opposite direction of the theorem is also true if we consider a tm-scheduler for $\mathsf{ex}(\mathcal{P})$ whose decisions are based on the $\mathcal{P}$-history only. We omit the details here as they require a Definition of $\mathcal{P}$-observation-based tm-schedulers and as they are similar to those in Section 4.3.

6.4 Chapter Summary

In this chapter, two kinds of button pushing experiments are examined, both having a direct relationship to failure trace equivalence (compare Proposition 6.1 on page 125 and Proposition 6.5 on page 146). From the point of view of Chapter 5, we use novel classes of schedulers, namely schedulers resolving external nondeterminism on the basis of the visible part of the process history. But here the “scheduler decision” is hidden behind the controllers of the weighted trace machine and the countdown timers of the delayed trace machine, respectively. These machine-based definitions are advantageous for the way a PTP is considered in this chapter: an open system using its visible actions to communicate with the environment. We fix the environment and observe the behavior of the processes to decide whether they respond in the same way or not.

Both relations are proven to be incomparable to the relations defined in Chapter 5 and reasons for their diverseness are given. Additionally, both relations fail to have the congruence property with respect to parallel composition. However, both are strictly coarser than phase type bisimulation.

Theorem 6.1 elucidates the relationship between $\mathcal{P}$ and $\mathsf{ex}(\mathcal{P})$. The fact that all representatives $s^{(j)} \in S_{\mathsf{ex}(\mathcal{P})}$ of a state $s \in S_{\mathcal{P}}$ inherit the action transitions of s is now clearly proven. Moreover, the "equivalence" of $\mathcal{P}$ and $\mathsf{ex}(\mathcal{P})$ is formalized by Theorem 6.1. The reason why the relationship between $\mathcal{P}$ and $\mathsf{ex}(\mathcal{P})$ is best clarified by using the delayed trace machine is that the operator ex takes into account that actions may be not be available immediately but after a certain amount of time.

We conclude this chapter with two remarks.

Remark 6.1
A closer look at the counterexamples, which prove that neither $=_{\mathsf{we}}$ (see Example 6.4 on page 132) nor $=_{\mathsf{de}}$ (see Example 6.7 on page 148) are congruences with respect to parallel composition, might suggest to consider an extension of the trace machine which has both action controllers and action countdown timers. But what is the intuitive meaning of setting the delay of an action timer to zero and the action's controller to a non-zero value? In case of weighted trace semantics we interpreted the weights as follows: If action a has weight w but action b has weight $2w$, the probability for b to be externally available is twice as high as that of a. For processes acting in continuous time, this expresses that b is provided more often (and thus within smaller time intervals) than a. But if the exact time instant is known at which a and b are provided, weights are needless even if both, a and b, are immediately available. In this case, it is an internal decision of the process which action is taken (a or b).

Remark 6.2
It is important to point out that the experimenter's interface that is used to determine the process' environment conditions is far from representing all the possible ways external stimuli are given. Consider, for instance, the

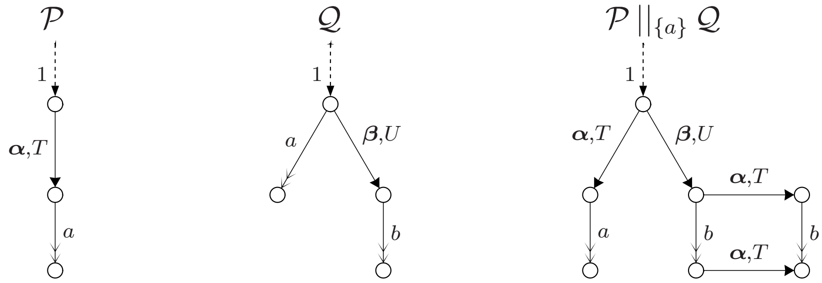

Figure 6.12: PTP $\mathcal{Q}$ constitutes an interaction environment for $\mathcal{P}$.

two processes $\mathcal{P}$ and $\mathcal{Q}$ in Figure 6.12. If $\mathcal{Q}$ and $\mathcal{P}$ synchronize over $A = \{a\}$ action b is offered only if the delay $X \sim F_{(\boldsymbol{\alpha},T)}$ of $\mathcal{P}$ is greater than delay $Y \sim F_{(\boldsymbol{\beta},U)}$ of $\mathcal{Q}$. No matter which controller position or countdown timer values are used, $\mathcal{Q}$ cannot be simulated by the machines defined so far. The environment's decision depends on the residence time of $\mathcal{P}$. But this motivates the idea of the next chapter in which we simulate the environment of a PTP *by another process instead of an experimenter. Obviously, the more environments we can use to test a process the more probable it becomes that the resulting equivalence is a congruence. However, this comes on the cost of a larger class of "tests" that have to be carried out. We conclude that the set of process environments that are simulated by the weighted/delayed trace machine is too small with regard to congruence properties.*

Chapter 7

Testing Semantics

7.1 Overview

Testing theory for concurrent processes is based on the seminal work of De Nicola and Hennessy [DH84]. A testing scenario for a process $\mathcal{P}$ is simulated by the parallel composition $\mathcal{P} \,||_{\mathsf{Act}}\, \mathcal{T}$ of P and a test process $\mathcal{T}$ (basically another process but equipped with a set of "success" states or actions). In the case of success states, $\mathcal{P}$ has "passed the test" if $\mathcal{P} \,||_{\mathsf{Act}}\, \mathcal{T}$ reaches a state for which the component corresponding to $\mathcal{T}$ is a success state. In the case of success actions, the successful pass of the test is determined by the execution of a success action. Two processes are *testing equivalent* if one passes a test if and only if the other one does.

In this chapter, we develop a testing theory for PTPs by extending the theory developed in [DH84] to our setting. In contrast to the previous chapter,

- the testing equivalence is restricted to SPTPs since parallel composition is only defined for SPTPs,
- we are no longer interested in trace observations but in the execution of a success action instead,
- since $\mathcal{T}$ branches probabilistically, we are now able to define that a set of external stimuli are given with a certain probability (compare Remark 6.2),

- we do not need to define a new classes of schedulers resolving the remaining nondeterminism of $\mathcal{P} \,||_{\mathsf{Act}}\, \mathcal{T}$ but can resort to the set HR.

We keep from $=_{\mathsf{we}}$ and $=_{\mathsf{de}}$ the following assumptions:

- SPTPs are regarded as open systems, i.e. we compare the success probabilities of two SPTPs for fixed environment conditions. As in the previous chapter we can informally say that $\mathcal{P}$ is equivalent to $\mathcal{Q}$ if for all environments E

  $$\text{Observations}(\mathcal{P} \text{ operates in } E) = \text{Observations}(\mathcal{Q} \text{ operates in } E).$$

  Here, E is represented by a test process $\mathcal{T}$ and an observation corresponds to a function which gives success probabilities.
- External nondeterminism may occur in the parallel composition $\mathcal{P} \,||_{\mathsf{Act}}\, \mathcal{T}$.
- The environment of the process under consideration is time-aware and probabilistic.
- We assume maximal progress, i.e. $\mathcal{P}$ cannot refuse interaction with $\mathcal{T}$ in stable states (this is formalized below).

For time-abstract frameworks, equivalences based on test processes enjoy several properties related to *congruence*. At least for internal, external and probabilistic choice as well as synchronous parallel, many relations are a congruence (e.g. compare [Seg96, KN98, Low93] and the references therein). An overview of the most important work on testing theory for time-abstract probabilistic models is given in [Wol04]. It includes a comparison and summary about [CSZ92, Chr90, Seg96, JY95, JY02, LS91]. Further interesting results for the time-abstract setting can be found amongst others in [Low93, SV03, KN98, KCS98]. For stochastic models acting in continuous time, only little work has been done in the area of testing: In [BC00] and the follow-up papers [Ber07, BB07] testing equivalences are defined for stochastic models without nondeterminism and with delays linked to actions. Synchronous parallel composition is achieved by assuming that the duration of an action is determined by exactly one of the two communication partners. Bernardo shows several congruence properties of a testing equivalence based on the simulation of a non-probabilistic environment (see [Ber07]). In our setting we have to deal with nondeterminism and the parallel composition operator for SPTPs is realized by synchronization on instantaneous transitions. Moreover, we deal with general tests (time-aware und probabilistic).

7.2 Testing Equivalence

In this thesis, we follow the idea of [Seg96] and use actions instead of states to report success. Let $\Theta = \{\theta_1, \theta_2, \ldots\}$ be a countable set of visible success actions with $\Theta \cap \mathsf{Act} = \emptyset$.

Definition 7.1 (Test Process)
A *test process* is a SPTP $\mathcal{T}$ over action set $\mathsf{Act}_\tau \cup \Theta$ such that

$$s \xrightarrow{\theta}\!\!\!\!\rightarrow s' \wedge \theta \in \Theta \implies s' \text{ is a deadlock state.}$$

The above definition stipulates that after success has been reported the experiment is over and, on the contrary, if $\mathcal{P}$ performs an infinite path while running in parallel with $\mathcal{T}$ it does not pass the test.
We synchronize over Act while testing a SPTP. Obviously, this assumption is appropriate since we are not interested in the visible behavior that $\mathcal{T}$ performs independently of $\mathcal{P}$. Furthermore, we assume a strengthening of the maximal progress assumption: the parallel composition is considered as *performance closed* and if a state in the compound process has an outgoing transition labeled by a visible action, this transition is enabled immediately. We remove all PH transitions of states that can execute an action transition. More precisely, let $\mathcal{P}|\mathcal{T}$ be the SPTP that is a copy of $\mathcal{P}\,||_{\mathsf{Act}}\,\mathcal{T}$ but for all states s of $\mathcal{P}\,||_{\mathsf{Act}}\,\mathcal{T}$ with $s \longrightarrow\!\!\!\!\rightarrow$ we remove all transitions $s \xrightarrow{\alpha, T} s'$.

Definition 7.2 (Testing Observation)
Let $\mathcal{P}$ be a SPTP, $\mathcal{T}$ a test process and $\mathcal{D}$ an HR-scheduler for $\mathcal{P}|\mathcal{T}$. A *testing*

observation is a function

$$\mathsf{te}_{\mathcal{P}}^{(\mathcal{T},\mathcal{D})} : \Theta \times \mathbb{R}_{\geq 0} \rightarrow [0,1]$$

such that

$$\begin{aligned} \mathsf{te}_{\mathcal{P}}^{(\mathcal{T},\mathcal{D})}(\theta, t) &= \mathsf{Pr}^{\mathcal{D}}(\{\pi \in \mathsf{path}(\mathcal{P}|\mathcal{T}) \mid \exists i, \exists \sigma \in \mathsf{Act}^* : trace(\pi{\downarrow_i}) = \sigma\theta, \\ & \quad time(\pi{\downarrow_i}) \leq t\}). \end{aligned}$$

The *set of testing observations* of $\mathcal{P}$ with respect to $\mathcal{T}$ is defined as

$$\mathsf{O}_{\mathsf{te}}(\mathcal{P}, \mathcal{T}) = \{\mathsf{te}_{\mathcal{P}}^{(\mathcal{T},\mathcal{D})} \mid \mathcal{D} \text{ is an } \mathsf{HR}\text{-scheduler for } \mathcal{P}|\mathcal{T}\}.$$

Definition 7.3 (Stochastic Testing Equivalence)
Two SPTPs $\mathcal{P}_1$ and $\mathcal{P}_2$ are *testing equivalent*, written $\mathcal{P}_1 =_{\mathsf{te}} \mathcal{P}_2$, iff for all test processes $\mathcal{T}$

$$\mathsf{O}_{\mathsf{te}}(\mathcal{P}_1, \mathcal{T}) = \mathsf{O}_{\mathsf{te}}(\mathcal{P}_2, \mathcal{T}).$$

Since the definition of the testing relation differs to that in [Seg96] only in that it is time-aware, the following proposition follows directly.

Proposition 7.1
Two probabilistic automata $\mathcal{P}_1$ and $\mathcal{P}_2$ are related by $=_{\mathsf{te}}$ iff they are testing equivalent with respect to [Seg96].

Our next objective is the relationship between $=_{\mathsf{te}}$ and the relations defined to far. Since $=_{\mathsf{te}}$ is only defined for SPTPs we restrict to this subclass of PTPs for the comparison.

Theorem 7.1
Let $\mathcal{P}$ and $\mathcal{Q}$ be SPTP*s. Then*

$$\mathcal{P} =_{\mathsf{te}} \mathcal{Q} \implies \mathcal{P} =_{\mathsf{de}} \mathcal{Q}.$$

Proof. Let us start with a counterexample that proves strictness. Consider Figure 6.10 on page 149. It holds $\mathcal{P}_1 \neq_{\mathsf{te}} \mathcal{P}_2$ if we assume a test process $\mathcal{T}$ which is a copy of $\mathcal{Q}$ but finally executes success actions θ_1 and θ_2 after trace ace and af, respectively (compare also Figure 6.11 on page 150 and Example 6.2 on page 125). From Example 6.2, $\mathcal{P}_1 =_{\mathsf{de}} \mathcal{Q}$.

We only give a proof sketch here and refer to Section A.3 of the appendix for the proof details. The idea is that given $\mathcal{P}_1 =_{\mathsf{te}} \mathcal{P}_2$, timer settings tm and tm-scheduler $\mathcal{F}_1$ a test process $\mathcal{T}_{\mathsf{tm}}$ is constructed which offers actions with the same probability as they are offered in the testing scenario with timer settings tm. We construct a scheduler $\mathcal{D} \in \mathsf{HR}(\mathcal{P}_1'|\mathcal{T}_{\mathsf{tm}})$ (where $\mathcal{P}_1'$ is a modified copy of $\mathcal{P}_1$) which resolves the remaining nondeterminism in $\mathcal{P}_1'|\mathcal{T}_{\mathsf{tm}}$ in the same way as $\mathcal{F}_1$ resolves nondeterminism of $\mathcal{P}_1$ acting under timer settings tm. Then we prove that trace observations of $\mathcal{P}_1$ with respect to tm and $\mathcal{F}_1$ have the same probability in $\mathcal{P}_1'|\mathcal{T}_{\mathsf{tm}}$ under scheduler $\mathcal{D}$. Test process $\mathcal{T}_{\mathsf{tm}}$ is enriched by success actions θ_σ which are used to report that trace σ has been performed (the enriched copy of $\mathcal{T}_{\mathsf{tm}}$ is denoted by $\mathcal{T}_{\mathsf{tm}}'$). From $\mathcal{P}_1 =_{\mathsf{te}} \mathcal{P}_2$ we get $\mathcal{P}_1' =_{\mathsf{te}} \mathcal{P}_2'$ (where $\mathcal{P}_2'$ is a modified copy of $\mathcal{P}_2$) and there exists a scheduler $\mathcal{E} \in \mathsf{HR}(\mathcal{P}_2'|\mathcal{T}_{\mathsf{tm}})$ such that

$$\mathsf{te}_{\mathcal{P}_1'}^{(\mathcal{T}_{\mathsf{tm}},\mathcal{D})}(\theta_\sigma, t) = \mathsf{te}_{\mathcal{P}_2'}^{(\mathcal{T}_{\mathsf{tm}},\mathcal{E})}(\theta_\sigma, t).$$

Finally, the same construction is used for $\mathcal{P}_2$ to define a tm-scheduler $\mathcal{F}_2$ which decides in the same way as $\mathcal{E}$ does in $\mathcal{P}_2|\mathcal{T}_{\mathsf{tm}}$ (that is, trace σ is performed with the same probability within $[0, t]$ for all $t \geq 0$) and derive

$$\mathsf{de}_{\mathcal{P}_1}^{(\mathsf{tm},\mathcal{F}_1)}(\sigma, t) = \mathsf{de}_{\mathcal{P}_2}^{(\mathsf{tm},\mathcal{F}_2)}(\sigma, t)$$

for all $\sigma \in \mathsf{Act}^*, t \geq 0$. □

The following proposition highlights the relationship between $=_{\mathsf{te}}$ and $=_{\mathsf{we}}$ as well as $=_{\mathsf{te}}$ and $=_{\mathsf{tr}}^{\mathsf{HR}}$ (on the set of all SPTPs).

Proposition 7.2

$=_{\mathsf{te}}$ *is incomparable to* $=_{\mathsf{we}}$ *and* $=_{\mathsf{tr}}^{\mathsf{HR}}$.

Proof.

$=_{\mathsf{we}} \not\subseteq =_{\mathsf{te}}$: Consider the two PTPs in Figure 6.4 on page 127. In Example 6.3 we proved that $\mathcal{P} =_{\mathsf{we}} \mathcal{Q}$. A test process that can be used to distinguish $\mathcal{P}$ and $\mathcal{Q}$ by $=_{\mathsf{te}}$ is constructed by modifying process $\mathcal{T}$ in Figure 6.4. We add transitions labeled by θ_{ac} and θ_{bf}, respectively, emerging from the two states which are reached by trace ac and bf, respectively. Now, consider a scheduler $\mathcal{D}_1 \in \mathsf{HR}(\mathcal{P}|\mathcal{T})$ which decides for the synchronous a-transition if $\mathcal{P}$ is in the left one of its initial states and for the synchronous b-transition if $\mathcal{P}$ is in the right one. Then for all $t \geq 0$

$$\mathsf{te}_{\mathcal{P}}^{(\mathcal{T},\mathcal{D}_1)}(\theta_{ac}, t) = 0.5 \text{ and } \mathsf{te}_{\mathcal{P}}^{(\mathcal{T},\mathcal{D}_1)}(\theta_{bf}, t) = 0.5.$$

A matching scheduler $\mathcal{D}_2 \in \mathsf{HR}(\mathcal{Q}|\mathcal{T})$ has to choose the a-transition in the left one of $\mathcal{Q}$'s initial states to match probability 0.5 for trace ac. But then $\mathsf{te}_{\mathcal{Q}}^{(\mathcal{T},\mathcal{D}_2)}(\theta_{bf}, t) \neq 0.5$. We conclude that no scheduler for $\mathcal{Q}|\mathcal{T}$ can match the observations under $\mathcal{D}_1$ and $\mathcal{P} \neq_{\mathsf{te}} \mathcal{Q}$ follows.

$=_{\mathsf{te}} \not\subseteq =_{\mathsf{we}}$: Consider Figure 6.7 on page 134. In Example 6.5, we proved that $\mathcal{P} \neq_{\mathsf{we}} \mathcal{Q}$. To see that $\mathcal{P} =_{\mathsf{te}} \mathcal{Q}$ we consider three kinds of test processes.

1. $\mathcal{T}$ initially offers no transition labeled by a and no PH transition followed by a. In this case, we get directly that $\mathcal{P}|\mathcal{T}$ and $\mathcal{Q}|\mathcal{T}$ perform success actions of $\mathcal{T}$ with the same probability. The only possibility to synchronize with $\mathcal{T}$ is the b-transition which is in both processes possible after two exponential delays with parameters -1 and -2.

2. $\mathcal{T}$ initially offers an a-transition (without any delay). Then both, $\mathcal{P}$ and $\mathcal{Q}$, immediately perform a without any delay. Again success actions of $\mathcal{T}$ are carried out with the same probability.

3. $\mathcal{T}$ offers an a-transition after a PH distributed delay X. Then both, $\mathcal{P}|\mathcal{T}$ and $\mathcal{Q}|\mathcal{T}$, perform an a if $X < Y$ where Y is the sum of two random variables exponentially distributed with parameters -1 and -2. Otherwise, if $X > Y$, both can perform b after Y time units (if b is offered by $\mathcal{T}$). Figure 6.8 on page 136 shows an example where a is offered after two exponential delays (with parameters -3 and -4). So, again we get the same success probabilities in $\mathcal{P}|\mathcal{T}$ and $\mathcal{Q}|\mathcal{T}$.

It is easy to see that all remaining kinds of test processes cannot be used to distinguish $\mathcal{P}$ and $\mathcal{Q}$. Hence, $\mathcal{P} =_{\mathsf{te}} \mathcal{Q}$.

$=^{\mathsf{HR}}_{\mathsf{tr}} \not\subseteq =_{\mathsf{te}}$: This can be seen by the counterexample in Figure 6.3 on page 126. Obviously, $\mathcal{P} =^{\mathsf{HR}}_{\mathsf{tr}} \mathcal{Q}$. In Example 6.2 on page 125 we argued that $\mathcal{P} =_{\mathsf{we}} \mathcal{Q}$ by defining appropriate controller settings ctr. For $=_{\mathsf{te}}$ we use a similar idea. If we put $\mathcal{P}$ and $\mathcal{Q}$ in parallel with process $\mathcal{T}$ (Figure 6.3, right) we get the same trace observations as $\mathcal{P}$ and $\mathcal{Q}$ under ctr, respectively. If we now assume that in $\mathcal{T}$ the two transitions labeled by e and f are both followed by a transition with label $\theta \in \Theta$, we can choose a scheduler for $\mathcal{P}|\mathcal{T}$ such that θ is performed with probability one (for all time bounds $t \geq 0$) but there is no scheduler for $\mathcal{Q}|\mathcal{T}$ such that θ executed with probability $p > 0.5$. Thus, $\mathcal{P} \neq_{\mathsf{te}} \mathcal{Q}$.

$=_{\mathsf{te}} \not\subseteq =^{\mathsf{HR}}_{\mathsf{tr}}$: A counterexample showing that $\mathcal{P} =_{\mathsf{te}} \mathcal{Q}$ does not imply $\mathcal{P} =^{\mathsf{HR}}_{\mathsf{tr}} \mathcal{Q}$ is given in Figure 6.7 on page 134. It holds that $\mathcal{P} =_{\mathsf{te}} \mathcal{Q}$ (compare case "$=_{\mathsf{te}} \not\subseteq =_{\mathsf{we}}$"). From Example 6.5 we know that $\mathcal{P} \neq^{\mathsf{HR}}_{\mathsf{tr}} \mathcal{Q}$ and hence

$=_{\mathsf{te}} \not\subseteq =_{\mathsf{tr}}^{\mathsf{HR}}.$

□

Note that the same counterexamples can be used to prove that $=_{\mathsf{te}}$ is incomparable to $=_{\mathsf{fa}}$ and $=_{\mathsf{re}}$.

Proposition 7.3
Bisimulation equivalence is strictly finer than testing equivalence, i.e. let $\mathcal{P}$ and $\mathcal{Q}$ be SPTP*s. Then*

$$\mathcal{P} =_{\mathsf{bs}} \mathcal{Q} \implies \mathcal{P} =_{\mathsf{te}} \mathcal{Q}.$$

Proof. The inclusion follows directly from Lemma 3.1 and the fact that $=_{\mathsf{bs}}$ is a congruence (see Proposition 4.1) because if we measure the probability that a certain success action is performed within $[0, t]$ we take the probability of the union of cylinder sets induced by sets Ξ_η (see Lemma 3.1).
For the opposite direction we refer to Figure 1.1. Obviously, $\mathcal{P}$ and $\mathcal{Q}$ are related by $=_{\mathsf{te}}$ but there is no bisimulation relation that can relate the initial distributions of $\mathcal{P}$ and $\mathcal{Q}$. There is no equivalent state to the initial one (ones) of $\mathcal{P}$ (of $\mathcal{Q}$). □

7.3 Chapter Summary

This chapter focused on the definition and classification of a testing equivalence for SPTPs. We relied mostly on Segala's testing approach [Seg96] and extended it to SPTPs. As expected, the testing relation turned out to be strictly finer than the trace relation and strictly coarser than bisimulation. The same classification holds for probabilistic automata. Moreover, the testing relation is also strictly finer than $=_{\mathsf{de}}$ since it is based on a more flexible class of testing scenarios. Strictness comes from the fact that $=_{\mathsf{de}}$ is no longer sensitive to different trace combinations if invisible transitions are

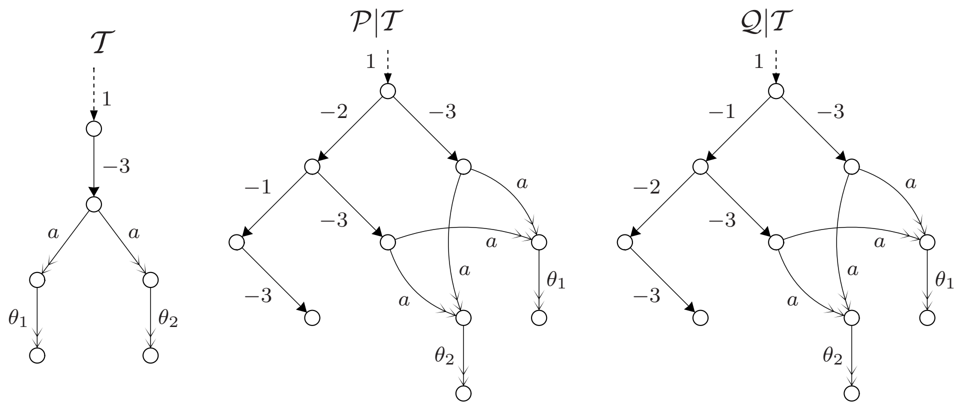

Figure 7.1: $\hat{\mathcal{P}} =_{\mathsf{bs}} \hat{\mathcal{Q}}$ but $\mathcal{P} = \mathsf{ex}(\hat{\mathcal{P}}) \neq_{\mathsf{te}} \mathcal{Q} = \mathsf{ex}(\hat{\mathcal{Q}})$

added (compare Figure 6.10 on page 149). But this distinction is crucial for the congruence property to hold which does so for $=_{\mathsf{te}}$.

Let us conclude with the question if $=_{\mathsf{te}}$ can be lifted on PTPs by the use of ex, i.e. let $\widetilde{=}_{\mathsf{te}}$ be the relation on PTPs that identifies $\mathcal{P}_1$ and $\mathcal{P}_2$ iff $\mathsf{ex}(\mathcal{P}_1) =_{\mathsf{te}} \mathsf{ex}(\mathcal{P}_2)$. We show that $=_{\mathsf{bs}} \not\subseteq \widetilde{=}_{\mathsf{te}}$ by considering the two SPTPs $\mathcal{P}$ and $\mathcal{Q}$ in Figure 6.7 on page 134. As explained in Example 6.5, we assume that $\mathcal{P} = \mathsf{ex}(\hat{\mathcal{P}})$ and $\mathcal{Q} = \mathsf{ex}(\hat{\mathcal{Q}})$. It holds that $\hat{\mathcal{P}} =_{\mathsf{bs}} \hat{\mathcal{Q}}$ because the PH transitions represent the same distribution. However, $\mathcal{P}$ and $\mathcal{Q}$ are not in $=_{\mathsf{te}}$ because they can be distinguished by test process $\mathcal{T}$ (compare Figure 7.1, left). To see this assume that an HR-scheduler $\mathcal{D}$ for $\mathcal{P}|\mathcal{T}$ chooses $\{1, 0\}$ in the upper right state with the two outgoing a-transitions and $\{0, 1\}$ in the lower left one (to reach the state with success action θ_1 and θ_2, respectively). The probability to perform θ_1 in $\mathcal{P}|\mathcal{T}$ is then given by

$$\mathsf{te}_{\mathcal{P}}^{(\mathcal{T},\mathcal{D})}(\theta_1, t) = \frac{3}{5}(1 - e^{-5t}) = \frac{3}{5} \cdot F_{-5}(t).$$

But this (sub-)distribution cannot be achieved by any HR-scheduler for $\mathcal{Q}|\mathcal{T}$ because the convolution of F_{-4} and F_{-5} (or a mixture of this convolution with F_{-4}) leads to different distributions (having different moments than $\frac{3}{5} \cdot F_{-5}(t)$) no matter which branching probabilities are chosen. Therefore, $\mathcal{P} \neq_{\mathsf{te}} \mathcal{Q}$.

Let us finally remark that it is possible to extend $=_{\mathsf{te}}$ on the set of PTPs by restricting to $\mathcal{P}$-observation-based schedulers. We leave this as future work and claim that this leads to a "natural" testing relation on PTPs.

Chapter 8

Conclusion

We have introduced the concept of phase type processes which form a very general class of models including, for instance, the class of labeled transition systems, probabilistic automata and interactive Markov chains. The main difference lies in the use of phase type transitions which are enabled after a phase type distributed delay. We considered time-aware schedulers to resolve nondeterministic choices based on the process history and made use of matrix operations based on the Kronecker product to give formal semantics in terms of path probabilities.

In the style of [Her02], a parallel composition operator for the subclass of single phase type processes has been presented which maintains the usual interleaving semantics. We have defined the ex-operator to show that this is not possible in the case of states having several phases. More precisely, there is, in general, no "natural" operator for the parallel composition of PTPs. The ex-operator provides the possibility to consider a PTP on a less abstract level. The phases each state has to pass through until it is left are added as extra states. The operator is useful in many respects because it sharpens the understanding of several problems related to phase type processes. It is important to point out that the operator applied to a PTP, say, $\mathcal{P}$ induces a class of $\mathcal{P}$-observation-based schedulers and that $\mathsf{ex}(\mathcal{P})$ should be analyzed with respect to this class. This restriction is necessary because otherwise the behavior of $\mathsf{ex}(\mathcal{P})$ may be different from that of $\mathcal{P}$ which leads to undesirable

$R_1 \setminus R_2$	$=_{\mathsf{tr}}^{\mathsf{HR}}$	$=_{\mathsf{we}}$	$=_{\mathsf{de}}$	$=_{\mathsf{te}}$	$=_{\mathsf{bs}}$
$=_{\mathsf{tr}}^{\mathsf{HR}}$	Def. 5.2	Fig. 6.3 Fig. 6.6	Fig. 6.3 Fig. 6.6	Fig. 6.3 Fig. 6.6	Fig. 5.2
$=_{\mathsf{we}}$	Fig. 6.4	Def. 6.3	Fig. 6.4 Fig. 6.6	Fig. 6.4 Fig. 6.6	Fig. 6.6
$=_{\mathsf{de}}$	Fig. 6.7	Fig. 6.7 Fig. 6.10	Def. 6.9	Fig. 6.10	Fig. 6.7
$=_{\mathsf{te}}$	Fig. 6.10	Fig. 6.10	$\subset$ Th. 7.1	Def. 7.3	Fig. 6.7
$=_{\mathsf{bs}}$	$\subset$ Prop. 5.2	$\subset$ Prop. 6.4	$\subset$ Prop. 6.7	$\subset$ Prop. 7.3	Def. 3.7

Table 8.1: The classification of the different notions of equivalence

effects on the notions of equivalence (for instance, $\mathsf{ex}(\mathcal{P})$ and $\mathcal{P}$ are not bisimilar). However, *with* this restriction we could establish Theorem 3.1 that proves the equality of $\mathsf{ex}(\mathcal{P})$ and $\mathcal{P}$. We leave as future work the further analysis of the ex-operator and $\mathcal{P}$-observation-based schedulers. Another interesting starting point for future work is the definition of an operator that transforms a SPTP into a PTP having less states by combining several successive PH transitions to a single one.

We also defined various relations to decide whether two PTPs are equivalent on a certain level of abstraction or not. In the case of trace equivalence, different notions arise by varying in the type of scheduler that serves to resolve the nondeterministic choices. We classified schedulers according to various criteria: time-aware vs. time-abstract ones, deterministic vs. randomized ones, history-dependent vs. stationary ones, and total vs. partial ones. Surprisingly, in most cases there is no correlation between the containment relation of the scheduler classes and the distinguishing power of the induced notions of trace equivalence. An overview of the results is given in Table 5.1 (see page 95).

For all remaining trace-based notions of equivalence we stick to time-abstract history-dependent schedulers and compared the different notions with respect to their distinguishing power as well. An overview is given in Table 8.1. It includes a comparison with bisimulation equivalence for PTPs which is, as opposed to the remaining relations, sensitive to τ-transitions. However, none of the counterexamples concerning $=_{\mathsf{bs}}$ make use of τ-transitions. Instead they rely on the fact that bisimulation-like relations are sensitive to the branching structure of the process whereas linear-time relations explore the possible execution sequences. The table is read as follows: Each entry corresponds to a comparison

$$\text{``}R_1 \overset{?}{\subset} R_2\text{''}$$

and it either contains one or two references to counterexamples or it contains the $\subset$-sign and the number of the theorem or proposition where it is shown. The entries of the diagonal refer to the definition of the relation.

The phase type process modeling paradigm raises a great variety of questions from which a significant part has been answered in this thesis.

Bibliography

[ALS84] David Assaf, Naftali A. Langberg, and Thomas H. Savits. Multivariate phase-type distributions. *Operations Research*, Vol. 32:688–701, 1984.

[ARM98] Adam Arkin, John Ross, and Harley H. McAdams. Stochastic kinetic analysis of developmental pathway bifurcation in phage lambda-infected escherichia coli cells. *Genetics*, 149(4):1633–48, 1998.

[Asm03] Søren Asmussen. *Applied Probability and Queues*. Springer, 2003.

[Bai98] Christel Baier. *On the Algorithmic Verification of Probabilistic Systems*. Habilitation, Universität Mannheim, 1998.

[BB87] Jos C. M. Baeten and Jan A. Bergstra. Ready-trace semantics for concrete process algebra with the priority operator. *The Computer Journal*, 30(6):498–506, 1987.

[BB07] Marco Bernardo and Stefania Botta. A survey of modal logics characterizing behavioral equivalences for nondeterministic and stochastic systems. *Mathematical Structures in Computer Science*, 2007+. To appear.

[BC00] Marco Bernardo and Rance Cleaveland. A theory of testing for Markovian processes. In *Proc. CONCUR 2000*, number 1877 in LNCS, pages 305–319. Springer, 2000.

[BCDK00] Peter Buchholz, Gianfranco Ciardo, Susanna Donatelli, and Peter Kemper. Complexity of memory-efficient kronecker operations with applications to the solution of markov models. *INFORMS J. on Computing*, 12(3):203–222, 2000.

[BD04] Mario Bravetti and Pedro R. D'Argenio. Tutte le algebre insieme: Concepts, discussions and relations of stochastic process algebras with general distributions. In C. Baier, B. Haverkort, H. Hermanns, J.-P. Katoen, and M. Siegle, editors, *Validation of Stochastic Systems*, volume 2925 of *Lecture Notes in Computer Science*, pages 44–88. Springer, 2004.

[Ber07] Marco Bernardo. Non-bisimulation-based markovian behavioral equivalences. *Journal of Logic and Algebraic Programming*, 2007+. To appear.

[BG96] Marco Bernardo and Roberto Gorrieri. Extended Markovian process algebra. In *Proc. CONCUR 1996*, number 1119 in LNCS, pages 315–330. Springer, 1996.

[BH97] Christel Baier and Holger Hermanns. Weak bisimulation for fully probabilistic processes. In *Proc. CAV 1997*, volume 1254 of *LNCS*, pages 119–130, 1997.

[BHHK03] Christel Baier, Boudewijn Haverkort, Holger Hermanns, and Joost-Pieter Katoen. Model-checking algorithms for continuous-time markov chains. *IEEE/ACM Transactions on Software Engineering*, 29(7), 2003.

[BHR84] Stephen D. Brookes, C. A. R. Hoare, and A. W. Roscoe. A theory of communicating sequential processes. *J. ACM*, 31(3):560–599, 1984.

[BKHW05] Christel Baier, Joost-Pieter Katoen, Holger Hermanns, and Verena Wolf. Comparative branching-time semantics for markov chains. *Inf. Comput.*, 200(2):149–214, 2005.

[Bre99] Pierre Bremaud. *Markov chains, Gibbs fields, Monte-Carlo simulation and queues.* Springer-Verlag, New York, 1999.

[Buc94] Peter Buchholz. Exact and ordinary lumpability in finite markov chains. *Journal of Applied Probability*, 31:59–75, 1994.

[BW82] Manfred Broy and Martin Wirsing. On the algebraic specification of finitary infinite communicating sequential processes. In *Proceedings of IFIP TC2 Working Conference on Formal Description of Programming Concepts II*, pages 171–196. North Holland Publ. Company, 1982.

[Chr90] Ivan Christoff. Testing equivalences and fully abstract models for probabilistic processes. In *Proceedings of CONCUR 1990*, number 458 in LNCS, pages 126–140. Springer, 1990.

[CM99] Christian Commault and Stephane Mocanu. Sparse representations of phase-type distributions. *Communications in Statistics – Stochastic Models*, 15(4):759–778, 1999.

[CSZ92] Rance Cleaveland, Scott Smolka, and Amy E. Zwarico. Testing preorders for probabilistic processes. In *Proceedings of ICALP 92*, number 623 in LNCS, pages 708–719. Springer, 1992.

[dA97] Luca de Alfaro. Formal verification of probabilistic systems. Phd thesis, Stanford University, 1997.

[Der70] Cyrus Derman. *Finite state Markovian decision processes.* Academic Press, New York, 1970.

[DH84] Rocco De Nicola and Matthew Hennessy. Testing equivalences for processes. *Theoretical Computer Science*, 34(1–2):83–133, 1984.

[ERKN99] Amani El-Rayes, Marta Kwiatkowska, and Gethin Norman. Solving infinite stochastic process algebra models through matrix-geometric methods. In J. Hillston and M. Silva, editors, *Proc. 7th Process Algebras and Performance Modelling Workshop (PAPM'99)*, pages 41–62. University of Zaragoza, 1999.

[Fel68] William Feller. *An Introduction to Probability Theory and its Applications, Vol. I.* John Wiley & Sons, Inc., 1968.

[FPS98] Paulo Fernandes, Brigitte Plateau, and William J. Stewart. Efficient descriptor-vector multiplications in stochastic automata networks. *Journal of the ACM*, 45(3):381–414, 1998.

[Gaj96] Magdalena Gajewsky. Kombination von stochastischen Prozessalgebren mit Phasentyp-Verteilungen. diploma thesis, TU Berlin, 1996.

[GHR92] Norbert Gotz, Ulrich Herzog, and Michae Rettelbach. Tipp – a language for timed processes and performance evaluation. report Technical Report 4/92, IMMD VII, University of Erlangen-Nurnberg, 1992.

[GS82] Geoffrey Grimmett and David Stirzaker. *Probability and Random Processes.* Oxford University Press, London, 1982.

[Hav98] Boudewijn R. Haverkort. *Performance of Computer Communication Systems: A Model-Based Approach.* John Wiley & Sons, Inc., 1998.

[Her02] Holger Hermanns. *Interactive Markov Chains: The Quest for Quantified Quality.*, volume 2428 of *LNCS*. Springer, 2002.

[Hil94] Jane Hillston. The nature of synchronisation. In *Proceedings of the Second International Workshop on Process Algebras and Performance Modelling*, pages 51–70, 1994.

[Hil96] Jane Hillston. *A Compositional Approach to Performance Modelling.* Cambridge University Press, 1996.

[HJ89] Hans Hansson and Bengt Jonsson. A framework for reasoning about time and reliability. In *IEEE Real-Time Systems Symposium*, pages 102–111, 1989.

[HK00] Holger Hermanns and Joost-Pieter Katoen. Automated compositional markov chain generation for a plain-old telephone system. *Sci. Comput. Program.*, 36(1):97–127, 2000.

[Hoa80] C. A. R. Hoare. A model for communicating sequential processes. In *On the Construction of Programs*, pages 229–254. 1980.

[HT92a] D.T. Huynh and L. Tian. On some equivalence relations for probabilistic processes. *Theoretical Computer Science*, 17(3):211–234, 1992.

[HT92b] Dung T. Huynh and Lu Tian. On some equivalence relations for probabilistic processes. *Fundam. Inform.*, 17(3):211–234, 1992.

[HZ05] Qi-Ming He and Hanqin Zhang. A note on unicyclic representations of phase type distributions. *Stochastic Models*, 21(2):465–483, 2005.

[HZ06] Qi-Ming He and Hanqin Zhang. Spectral polynomial algorithms for computing bi-diagonal representations for phase type distributions and matrix-exponential distributions. *Stochastic Models*, 22(2):289–317, 2006.

[JL96] Chris Jedrzycki and Victor C. M. Leung. Probability distribution of channel holding time in cellular telephony systems. In

Proceedings of the 46th IEEE Vehicular Technology Conference, 1996.

[JS90] Chi-Chang Jou and Scott A. Smolka. Equivalences, congruences, and complete axiomatizations for probabilistic processes. In *Proceedings on CONCUR*, pages 367–383, New York, NY, USA, 1990. Springer-Verlag New York, Inc.

[JT88] Mary A. Johnson and Michael R. Taaffe. The denseness of phase distributions. Research memorandum, Purdue University, 1988.

[JY95] Bengt Jonsson and Wang Yi. Compositional testing preorders for probabilistic processes. In *Proc. 10th IEEE Int. Symp. on Logic in Computer Science*, pages 431–441, 1995.

[JY02] Bengt Jonsson and Wang Yi. Testing preorders for probabilistic processes can be characterized by simulations. *Theor. Comput. Sci.*, 282(1):33–51, 2002.

[KCS98] K. Narayan Kumar, Rance Cleaveland, and Scott Smolka. Infinite probabilistic and non-probabilistic testing. In *Proceedings of the 18th FST and TCS Conference*, number 1530 in LNCS, pages 209–220. Springer, 1998.

[KEBC05] Mads Kœrn, Timothy C. Elston, William J. Blake, and James J. Collins. Stochasticity in gene expression: From theories to phenotypes. *Nat Rev Genet*, 6(6):451–464, June 2005.

[KN98] Marta Kwiatkowska and Gethin Norman. A testing equivalence for reactive probabilistic processes. In *Proc. EXPRESS 1998*, ENTCS, 1998.

[Low93] Gawin Lowe. Representing nondeterminism and probabilistic behaviour in reactive processes. Technical Report Technical Report TR-11-93, Oxford University, 1993.

[LR99a] Guy Latouche and Vaidyanat Ramaswami. *Introduction to Matrix Analytic Methods in Stochastic Modeling.* ASA-SIAM, Philadelphia, 1999.

[LR99b] Guy Latouche and Vaidyanat Ramaswami. *Introduction to Matrix Analytic Methods in Stochastic Modelling.* ASA & SIAM, 1999.

[LS91] Kim G. Larsen and Arne Skou. Bisimulation through probabilistic testing. *Information and Computation*, 94:1–28, 1991.

[Mil80] Robin Milner. *A Calculus of Communicating Systems.* Lecture Notes in Computer Science Vol. 92. Springer-Verlag, 1980.

[Mil89] Robin Milner. *Communication and concurrency.* Prentice-Hall, Inc., 1989.

[MSZ00] Avishai Mandelbaum, Anat Sakov, and Sergey Zeltyn. Empirical analysis of a call center. Technical report, Technion, Israel Institute of Technology, 2000.

[Neu81] Marcel Neuts. *Matrix-geometric solutions in stochastic models.* John Hopkins University Press, Baltimore, 1981.

[O'C99] Colm Art O'Cinneide. Phase-type distributions: Open problems and a few properties. *Stochastic Models*, 15(4):731–757, 1999.

[OH83] Ernst-Rüdiger Olderog and C. A. R. Hoare. Specification-oriented semantics for communicating processes. In *Proceedings of ICALP*, pages 561–572, 1983.

[PF95] Vern Paxson and Sally Floyd. Wide area traffic: the failure of Poisson modeling. *IEEE/ACM Transactions on Networking*, 3(3):226–244, 1995.

[Phi87] Iain Phillips. Refusal testing. *Theor. Comput. Sci.*, 50(3):241–284, 1987.

[Pla84] Brigitte Plateau. De l'evaluation du paralllisme et de la synchronisation. Phd thesis, Universit de Paris XII, Orsay (France), 1984.

[PLS00] Anna Philippou, Insup Lee, and Oleg Sokolsky. Weak bisimulation for probabilistic systems. *Lecture Notes in Computer Science*, 1877:334+, 2000.

[Pnu85] Amir Pnueli. Linear and branching structures in the semantics and logics of reactive systems. In *Proceedings of the 12th Colloquium on Automata, Languages and Programming*, pages 15–32, London, UK, 1985. Springer-Verlag.

[Pri95] Corrado Priami. Stochastic pi-calculus. *The Computer Journal*, 38(7):578–589, 1995.

[Put94] Martin L. Puterman. *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. John Wiley & Sons, 1994.

[Seg95] Roberto Segala. Modeling and verification of randomized distributed real-time systems. Phd thesis, Massachusetts Institute of Technology, 1995.

[Seg96] Roberto Segala. Testing probabilistic automata. In *Proceedings of CONCUR 1996*, volume 1119 of *Lecture Notes in Computer Science*, pages 299–314. Springer, 1996.

[SL95] Roberto Segala and Nancy Lynch. Probabilistic simulations for probabilistic processes. *Nordic Journal of Computing*, 2(2):250–273, 1995.

[Ste95] William J. Stewart. *Introduction to the Numerical Solution of Markov Chains*. Princeton University Press, 1995.

[Sto02] Mariëlle I.A. Stoelinga. *Alea jacta est: verification of probabilistic, real-time and parametric systems.* PhD thesis, University of Nijmegen, the Netherlands, April 2002.

[SV03] Mariëlle I.A. Stoelinga and Fritz W. Vaandrager. A testing scenario for probabilistic automata. In *Proc. of ICALP 2003*, volume 2719 of *LNCS*, pages 407–418. Springer, 2003.

[Var85] Moshe Y. Vardi. Automatic verification of probabilistic concurrent finite-state programs. In *Annual Symposium on Foundations of Computer Science (FOCS)*, pages 327–338. IEEE CS Press, 1985.

[vGla90] Rob J. van Glabbeek. The linear time-branching time spectrum. In *Proceedings on Theories of Concurrency: Unification and Extension*, pages 278–297. Springer-Verlag, 1990.

[vG93] Rob J. van Glabbeek. The linear time - branching time spectrum II. In *Proceedings of the 4th International Conference on Concurrency Theory*, volume 715 of *LNCS*, pages 66–81. Springer-Verlag, 1993.

[vGSST90] Rob J. van Glabbeek, S. A. Smolka, B. Steffen, and C. Tofts. Reactive, generative, and stratified models of probabilistic processes. In *Logic in Computer Science*, pages 130–141, 1990.

[WBM06] Verena Wolf, Christel Baier, and Mila Majster-Cederbaum. Trace semantics for stochastic systems with nondeterminism. In *Proc. of the fourth Workshop on Quantitative Aspects of Programming (QAPL)*, volume 164 of *ENTCS*, pages 187–204, 2006.

[WM05] Olaf Wolkenhauer and Mihajlo Mesarović. Feedback dynamics and cell function: Why systems biology is called systems biology. *Molecular BioSystems*, 1(1):14–16, 2005.

[WMB05] Verena Wolf, Mila Majster-Cederbaum, and Christel Baier. Trace machines for observing continuous time Markov chains. In *Proc. of the third Workshop on Quantitative Aspects of Programming Languages (QAPL)*, volume 153 of *ENTCS*, pages 259–277, 2005.

[Wol04] Verena Wolf. Testing theory for probabilistic systems. In Manfred Broy, Bengt Jonsson, Joost-Pieter Katoen, Martin Leucker, and Alexander Pretschner, editors, *Model-Based Testing of Reactive Systems*, volume 3472 of *Lecture Notes in Computer Science*, pages 233–275. Springer, 2004.

Appendix A

Proofs

A.1 Proof of Proposition 6.3

Proposition A.1

Let $\mathcal{P}_1$ and $\mathcal{P}_2$ be PTP*s. $\mathcal{P}_1 =_{\mathsf{we}} \mathcal{P}_2$ implies $\mathcal{P}_1 =^{\mathsf{TR}}_{\mathsf{tr}} \mathcal{P}_2$.*

Proof. Assume that $\mathcal{P}_1 =_{\mathsf{we}} \mathcal{P}_2$. Now, let $\mathcal{D}_1$ be a trace dependent scheduler for $\mathcal{P}_1$ and let $\sigma \in \mathsf{Act}^*$. According to Defintion 6.5 there exist values $z_\perp, z_a, z_b, \ldots \in \mathbb{R}_{\geq 0}$ such that the above conditions are fulfilled. Since these values depend on σ and $\mathcal{D}_1$ we might sometimes write $z_a^{\mathcal{D}_1,\sigma}$ instead of z_a and $z_\perp^{\mathcal{D}_1,\sigma}$ instead of $z_\perp$ in the remainder of the proof. We perform a button pushing experiment using the weighted trace machine and assume controller settings $\mathsf{ctr}_\sigma(a) := z_a = z_a^{\mathcal{D}_1,\sigma}$ for $a \in \mathsf{Act}$, $\mathsf{ctr}_\sigma(\perp) := z_\perp = z_\sigma^{\mathcal{D}_1}(\perp)$. Then for each path fragment ξ with $trace(\xi) = \sigma$ the normalization constant $N(\xi, \mathsf{ctr})$ which is used to calculate the probability of a weighted trace observation equals

$$N(\xi, \mathsf{ctr}) = \mathsf{ctr}_\sigma(\perp) + \sum_{\substack{a \in \mathsf{Act} \\ last(\xi) \xrightarrow{a}}} \mathsf{ctr}_\sigma(a) = z_\perp + \sum_{\substack{a \in \mathsf{Act} \\ last(\xi) \xrightarrow{a}}} z_a = N.$$

The idea of the proof is to construct a trace dependent scheduler $\mathcal{D}_2$ for $\mathcal{P}_2$

such that for all $t \geq 0$ and all $\sigma \in \mathsf{Act}^*$

$$\mathsf{tr}_{\mathcal{P}_1}^{\mathcal{D}_1}(\sigma, t) = \mathsf{tr}_{\mathcal{P}_2}^{\mathcal{D}_2}(\sigma, t). \tag{A.1}$$

This is done in two steps. First we construct an internal scheduler $\mathcal{E}_1$ for $\mathcal{P}_1$ based on the decisions of $\mathcal{D}_1$. Since $\mathcal{P}_1 =_{\mathsf{we}} \mathcal{P}_2$ there exists a matching internal scheduler $\mathcal{E}_2$ for $\mathcal{P}_2$. Based on $\mathcal{E}_2$ we construct $\mathcal{D}_2$ and show that Equation A.1 holds.

i) An internal scheduler $\mathcal{E}_1$ is defined as follows: If $last(\xi)$ is stable we let, for all $a \in \mathsf{Act}$,

$$\mathcal{E}_1(\xi)(a, \mu) \quad := \quad \tfrac{\mathcal{D}_1(\xi)(a,\mu)}{\mathcal{D}_1(\xi,a)}$$

if $\mathcal{D}_1(\xi)(a, \mu) > 0$ and $\mathcal{E}_1(\xi)(a, \mu) := 0$ otherwise. But then if $\mathcal{D}_1(\xi, a) > 0$ we have that

$$\begin{aligned} \mathcal{D}_1(\xi)(a, \mu) \quad &= \quad \mathcal{D}_1(\xi, a) \cdot \tfrac{\mathcal{D}_1(\xi)(a,\mu)}{\mathcal{D}_1(\xi,a)} \\ &\overset{\text{Def. 6.5}}{=} \quad \tfrac{z_a}{N} \cdot \mathcal{E}_1(\xi)(a, \mu) \\ &= \quad \tfrac{\mathsf{ctr}_\sigma(a)}{N(\xi,\mathsf{ctr})} \cdot \mathcal{E}_1(\xi)(a, \mu). \end{aligned} \tag{A.2}$$

If $last(\xi)$ is unstable and $N = N(\xi, \mathsf{ctr}) > 0$, we have from Definition 6.5 values p_a^ξ for each $a \in \mathsf{Act}$ with $last(\xi) \xrightarrow{a}\!\!\!\rightarrow$. For $\mu \in \mathsf{dis}(S_1)$, $a \in \mathsf{Act}$ we put

$$\mathcal{E}_1(\xi)(a, \mu) \quad := \quad \begin{cases} \tfrac{\mathcal{D}_1(\xi)(a,\mu)}{\mathcal{D}_1(\xi,a)} \cdot p_a^\xi & \text{if } \mathcal{D}_1(\xi)(a, \mu) > 0, \\ 0 & \text{otherwise.} \end{cases}$$

This yields for $\mathcal{D}_1(\xi)(a, \mu) > 0$ that

$$\begin{aligned} \mathcal{D}_1(\xi)(a, \mu) \quad &= \quad \mathcal{D}_1(\xi, a) \cdot \tfrac{\mathcal{D}_1(\xi)(a,\mu)}{\mathcal{D}_1(\xi,a)} \\ &\overset{\text{Def. 6.5}}{=} \quad \tfrac{z_a}{N} \cdot p_a^\xi \cdot \tfrac{\mathcal{D}_1(\xi)(a,\mu)}{\mathcal{D}_1(\xi,a)} \\ &= \quad \tfrac{\mathsf{ctr}_\sigma(a)}{N(\xi,\mathsf{ctr})} \cdot \mathcal{E}_1(\xi)(a, \mu). \end{aligned} \tag{A.3}$$

Finally, we set

$$\mathcal{E}_1(\xi)(\tau,\mu) \;:=\; \begin{cases} \frac{\mathcal{D}_1(\xi)(\tau,\mu)}{\mathcal{D}_1(\xi,\tau)} & \text{if } \mathcal{D}_1(\xi)(\tau,\mu) > 0, \\ 0 & \text{otherwise.} \end{cases}$$

If $\mathcal{D}_1(\xi,\tau) > 0$ and $N > 0$ we get with $\mathcal{E}_1(\xi) =: \vartheta$

$$\begin{aligned} \mathcal{D}_1(\xi)(\tau,\mu) &= \mathcal{D}_1(\xi,\tau)\cdot\frac{\mathcal{D}_1(\xi)(\tau,\mu)}{\mathcal{D}_1(\xi,\tau)} \\ &\overset{\text{Def. 6.5}}{=} \Big(\frac{z_\perp}{N} + \sum_{\substack{a\in\mathsf{Act} \\ last(\xi)\xrightarrow{a}\!\!\!\rightarrow}} \frac{z_a}{N}\cdot(1-p_a^\xi)\Big)\cdot\mathcal{E}_1(\xi)(\tau,\mu) \\ &= \Big(\frac{\mathsf{ctr}_\sigma(\perp)}{N(\xi,\mathsf{ctr})} + \sum_{a\in\mathsf{Act}} \frac{\mathsf{ctr}_\sigma(a)}{N(\xi,\mathsf{ctr})}\cdot\vartheta_a^\perp\Big)\cdot\mathcal{E}_1(\xi)(\tau,\mu) \\ &= \big(p_e + p_i\big)\cdot\mathcal{E}_1(\xi)(\tau,\mu) \end{aligned} \tag{A.4}$$

where p_i and p_e are defined as in Equation (6.2) (see page 120). If $N = N(\xi,\mathsf{ctr}) = 0$ then $\mathcal{D}_1(\xi,\tau) = 1$ which yields $\mathcal{E}_1(\xi)(\tau,\mu) = \mathcal{D}_1(\xi)(\tau,\mu)$ for all μ and $p_e = 1, p_i = 0$.

It is easy to check that $\mathcal{E}_1$ is an internal scheduler (compare Definition 6.1 on page 118). Moreover, with the Equations (A.2)-(A.4) we get that

$$\begin{aligned} &a)\quad \pi\in\mathsf{path}(\mathcal{P}) \text{ is a } \mathcal{D}_1\text{-path iff } \pi \text{ is a } (\mathsf{ctr},\mathcal{E}_1)\text{-path.} \\ &b)\quad \mathsf{Pr}^{(\mathsf{ctr},\mathcal{E}_1)} = \mathsf{Pr}^{\mathcal{D}_1} \end{aligned} \tag{A.5}$$

where statement b) can be shown by induction on the length k of cylinder sets $C_\kappa(J_1,\ldots,J_k)$ of $\mathcal{D}_1$-paths.

ii) Let $\mathcal{E}_2$ be an internal scheduler for $\mathcal{P}_2$ such that $\mathsf{we}_{\mathcal{P}_1}^{(\mathsf{ctr},\mathcal{E}_1)} = \mathsf{we}_{\mathcal{P}_2}^{(\mathsf{ctr},\mathcal{E}_2)}$. We construct a trace dependent scheduler $\mathcal{D}_2 \in \mathsf{HR}(\mathcal{P}_2)$ as follows:

Let $\sigma = trace(\xi)$, and $\mu \in \mathsf{dis}_{S_2}$. If $last(\xi)$ is stable we define for all $a \in \mathsf{Act}$

$$\mathcal{D}_2(\xi)(a,\mu) := \begin{cases} \frac{\mathsf{ctr}_\sigma(a)}{N(\xi,\mathsf{ctr})}\cdot\mathcal{E}_2(\xi)(a,\mu) & \text{if } N(\xi,\mathsf{ctr}) > 0, \\ 0 & \text{otherwise.} \end{cases}$$

If $last(\xi)$ is unstable and $\mathcal{E}_2(\xi) =: \vartheta$, the probability that $\mathcal{D}_2$ decides for $a \in \mathsf{Act}$ is defined by

$$\mathcal{D}_2(\xi, a) := \begin{cases} \frac{\mathsf{ctr}_\sigma(a)}{N(\xi,\mathsf{ctr})} \cdot (1 - \vartheta_a^\perp) & \text{if } N(\xi, \mathsf{ctr}) > 0, \\ 0 & \text{otherwise.} \end{cases}$$

Furthermore, if $\vartheta_a(\mu) > 0$ we put

$$\mathcal{D}_2(\xi)(a, \mu) \;:=\; \mathcal{D}_2(\xi, a) \cdot \tfrac{\vartheta_a(\mu)}{1-\vartheta_a^\perp},$$

and $\mathcal{D}_2(\xi)(a, \mu) = 0$ otherwise. Thus,

$$\begin{aligned} \mathcal{D}_2(\xi)(a, \mu) &= \mathcal{D}_2(\xi, a) \cdot \tfrac{\vartheta_a(\mu)}{1-\vartheta_a^\perp} \\ &= \tfrac{\mathsf{ctr}_\sigma(a)}{N(\xi,\mathsf{ctr})} \cdot \vartheta_a(\mu) \\ &= \tfrac{\mathsf{ctr}_\sigma(a)}{N(\xi,\mathsf{ctr})} \cdot \mathcal{E}_2(\xi)(a, \mu). \end{aligned} \tag{A.6}$$

In the case that $last(\xi) =: s$ is unstable we let

$$\mathcal{D}_2(\xi, \tau) := \begin{cases} \frac{\mathsf{ctr}_\sigma(\perp)}{N(\xi,\mathsf{ctr})} + \sum\limits_{\substack{a \in \mathsf{Act}, \\ s \xrightarrow{a}\!\!\!\to}} \frac{\mathsf{ctr}_\sigma(a)}{N(\xi,\mathsf{ctr})} \cdot \vartheta_a^\perp & \text{if } N(\xi, \mathsf{ctr}) > 0, \\ 1 & \text{otherwise} \end{cases}$$

and for $\vartheta_\tau(\mu) > 0$

$$\mathcal{D}_2(\xi)(\tau, \mu) \;:=\; \mathcal{D}_2(\xi, \tau) \cdot \vartheta_\tau(\mu),$$

and $\mathcal{D}_2(\xi)(\tau, \mu) = 0$ otherwise.

It holds that $\mathcal{D}_2 \in \mathsf{TR}$. To see this, assume that for all $\sigma \in \mathsf{Act}^*$ we choose $z_a^{\mathcal{D}_2,\sigma} := z_a^{\mathcal{D}_1,\sigma}$ for $a \in \mathsf{Act}$ and $z_\perp^{\mathcal{D}_2,\sigma} := z_\perp^{\mathcal{D}_1,\sigma}$ in Definition 6.5 (and get the same normalization constant N). Then, if ξ is stable and $N = N(\xi, \mathsf{ctr}) > 0$, we calculate

$$\mathcal{D}_2(\xi, a) = \sum_\mu \mathcal{D}_2(\xi)(a, \mu) = \frac{\mathsf{ctr}_\sigma(a)}{N(\xi, \mathsf{ctr})} = \frac{z_a^{\mathcal{D}_1,\sigma}}{N} = \frac{z_a^{\mathcal{D}_2,\sigma}}{N}$$

and

$$\begin{aligned}
\mathcal{D}_2(\xi)^{\perp} &= 1 - \sum_{\substack{a \in \mathsf{Act} \\ s \xrightarrow{a}\!\!\!\rightarrow}} \mathcal{D}_2(\xi, a) \\
&= 1 - \sum_{\substack{a \in \mathsf{Act} \\ s \xrightarrow{a}\!\!\!\rightarrow}} \frac{\mathsf{ctr}_\sigma(a)}{N(\xi, \mathsf{ctr})} \\
&= \frac{\mathsf{ctr}_\sigma(\perp)}{N(\xi, \mathsf{ctr})} = \frac{z_{\perp}^{\mathcal{D}_1,\sigma}}{N} = \frac{z_{\perp}^{\mathcal{D}_2,\sigma}}{N}.
\end{aligned}$$

If ξ is unstable and $last(\xi) = s \xrightarrow{a}\!\!\!\rightarrow$, we set $p_a^{\xi} := 1 - \vartheta_a^{\perp}$ where $\vartheta := \mathcal{E}_2(\xi)$. If $N = 0$ we have $\mathcal{D}_2(\xi, \tau) = 1$. If $N > 0$ we calculate

$$\begin{aligned}
\mathcal{D}_2(\xi, \tau) &= \frac{\mathsf{ctr}_\sigma(\perp)}{N(\xi, \mathsf{ctr})} + \sum_{\substack{a \in \mathsf{Act} \\ s \xrightarrow{a}\!\!\!\rightarrow}} \frac{\mathsf{ctr}_\sigma(a)}{N(\xi, \mathsf{ctr})} \cdot \vartheta_a^{\perp} \\
&= \frac{z_{\perp}^{\mathcal{D}_1,\sigma}}{N} + \sum_{\substack{a \in \mathsf{Act} \\ s \xrightarrow{a}\!\!\!\rightarrow}} \frac{z_a^{\mathcal{D}_1,\sigma}}{N} \cdot (1 - p_a^{\xi}) \\
&= \frac{z_{\perp}^{\mathcal{D}_2,\sigma}}{N} + \sum_{\substack{a \in \mathsf{Act} \\ s \xrightarrow{a}\!\!\!\rightarrow}} \frac{z_a^{\mathcal{D}_2,\sigma}}{N} \cdot (1 - p_a^{\xi})
\end{aligned}$$

and

$$\begin{aligned}
\mathcal{D}_2(\xi, a) &= \frac{\mathsf{ctr}_\sigma(a)}{N(\xi, \mathsf{ctr})} \cdot (1 - \vartheta_a^{\perp}) \\
&= \frac{z_a^{\mathcal{D}_1,\sigma}}{N} \cdot p_a^{\xi} = \frac{z_a^{\mathcal{D}_2,\sigma}}{N} \cdot p_a^{\xi}.
\end{aligned}$$

Therefore, the conditions of Definition 6.5 hold and $\mathcal{D}_2 \in \mathsf{TR}$.

Similar as for $\mathcal{D}_1$ and $\mathcal{E}_1$, the following statements can be derived from the definition of $\mathcal{D}_2$ and Equation (A.6):

$$\begin{aligned}
&a) \quad \pi \in \mathsf{path}(\mathcal{P}) \text{ is a } \mathcal{D}_2\text{-path iff } \pi \text{ is a } (\mathsf{ctr}, \mathcal{E}_2)\text{-path.} \\
&b) \quad \mathsf{Pr}^{(\mathsf{ctr}, \mathcal{E}_2)} = \mathsf{Pr}^{\mathcal{D}_2}
\end{aligned} \qquad \text{(A.7)}$$

where $b)$ requires an induction on the length k of cylinder sets $C_\kappa(J_1, \ldots, J_k)$ of $\mathcal{D}_2$-paths.

It remains to show that $\mathsf{tr}_{\mathcal{P}_1}^{\mathcal{D}_1} = \mathsf{tr}_{\mathcal{P}_2}^{\mathcal{D}_2}$. Let $\sigma \in \mathsf{Act}^*, t \geq 0$. Then

$$\begin{aligned}
\mathsf{tr}_{\mathcal{P}_1}^{\mathcal{D}_1}(\sigma, t) &= \mathsf{Pr}^{\mathcal{D}_1}(\{\pi \in \mathsf{path}(\mathcal{P}_1) \mid \pi \text{ is a } \mathcal{D}_1\text{-path and} \\
&\qquad \exists i : trace(\pi{\downarrow_i}) = \sigma, time(\pi{\downarrow_i}) \leq t\}) \\
&\overset{\text{Eq. A.5}}{=} \mathsf{Pr}^{(\mathsf{ctr},\mathcal{E}_1)}(\{\pi \in \mathsf{path}(\mathcal{P}_1) \mid \pi \text{ is a } (\mathsf{ctr},\mathcal{E}_1)\text{-path and} \\
&\qquad \exists i : trace(\pi{\downarrow_i}) = \sigma, time(\pi{\downarrow_i}) \leq t\}) \\
&= \mathsf{we}_{\mathcal{P}_1}^{(\mathsf{ctr},\mathcal{E}_1)} \\
&= \mathsf{we}_{\mathcal{P}_2}^{(\mathsf{ctr},\mathcal{E}_2)} \\
&= \mathsf{Pr}^{(\mathsf{ctr},\mathcal{E}_2)}(\{\pi \in \mathsf{path}(\mathcal{P}_2) \mid \pi \text{ is a } (\mathsf{ctr},\mathcal{E}_2)\text{-path and} \\
&\qquad \exists i : trace(\pi{\downarrow_i}) = \sigma, time(\pi{\downarrow_i}) \leq t\}) \\
&= \mathsf{Pr}^{\mathcal{D}_2}(\{\pi \in \mathsf{path}(\mathcal{P}_2) \mid \pi \text{ is a } \mathcal{D}_2\text{-path and} \\
&\qquad \exists i : trace(\pi{\downarrow_i}) = \sigma, time(\pi{\downarrow_i}) \leq t\}) \\
&= \mathsf{tr}_{\mathcal{P}_2}^{\mathcal{D}_2}(\sigma, t).
\end{aligned}$$

This proves that $\mathcal{P}_1 =_{\mathsf{we}} \mathcal{P}_2$ implies $\mathcal{P}_1 =_{\mathsf{tr}}^{\mathsf{TR}} \mathcal{P}_2$. □

A.2 Proof of Theorem 6.1

Theorem A.1

Let $\mathcal{P}$ be a PTP *and* $\mathsf{tm} : \mathsf{Act}^* \times \mathsf{Act} \to \big(\mathcal{R} \cup \{\mathsf{I}, \mathsf{B}\}\big)$. *Then for each* tm-*scheduler $\mathcal{F}'$ for $\mathcal{P}$ there exists a* tm-*scheduler $\mathcal{F}$ for* $\mathsf{ex}(\mathcal{P})$ *such that for all* $\mathsf{C}_\zeta \in \mathcal{C}_{\mathcal{P}}^{\mathsf{tm}}$

$$\mathsf{Pr}_{\mathcal{P}}^{(\mathsf{tm},\mathcal{F}')}(\mathsf{C}_\zeta) = \mathsf{Pr}_{\mathsf{ex}(\mathcal{P})}^{(\mathsf{tm},\mathcal{F})}(\mathsf{ex}(\mathsf{C}_\zeta)).$$

Proof. We construct $\mathcal{F}$ as follows: Let $\xi \in \mathsf{pathf}(\mathsf{ex}(\mathcal{P}), \mathsf{tm})$ and let ξ' be the maximal prefix of ξ that is $\mathcal{P}$-observable. Then

$$\mathcal{F}(\xi)(a, \mu) := \mathcal{F}'(\mathsf{contr}(\xi'))(a, \mu') \tag{A.8}$$

where $\mu'(u) \cdot \gamma_u(j) = \mu(u^{(j)})$ for all u, $j \in \{1, \ldots, n_u\}$ (compare also Definition 4.2 on page 71). $\mathcal{F}$ is a tm-scheduler because, if ξ' is the maximal prefix of ξ that is $\mathcal{P}$-observable and $last(\xi) = (s^{(j)}, \mathsf{ph})$, $last(\xi') = (v^{(i)}, \mathsf{ph}')$ then

- ⋄ $s = v$,
- ⋄ $s^{(j)} \xrightarrow{a} \mu$ if and only if $s \xrightarrow{a} \mu'$ where $\mu'(u) \cdot \gamma_u(k) = \mu(u^{(k)})$ for all u, $k \in \{1, \ldots, n_u\}$,
- ⋄ $(s^{(j)}, \mathsf{ph})$ is stable if and only if $(s, \mathsf{ph}') = last(\mathsf{contr}(\xi'))$ is stable.

We now proceed by induction on the length k of ζ. For timer settings tm_σ, $\sigma \in \mathsf{Act}^*$, the probability of timer initialization ph is given by $\mathsf{Pr}(\mathsf{ph}, \sigma)$ (compare Equation 6.4 on page 142). Let $k = 1$ and let $\mathsf{C}_\zeta = \mathsf{C}_{(s,\mathsf{ph})}$, i.e., $\mathsf{C}_{(s,\mathsf{ph})}$ is the set of all tm-paths starting with (s, ph). Then $\mathsf{ex}(\mathsf{C})$ is the set of all tm-paths in $\mathsf{ex}(\mathcal{P})$ starting with $(s^{(i)}, \mathsf{ph})$, $i \in \{1, \ldots, n_s\}$ and

$$\begin{aligned}
\mathsf{Pr}_{\mathcal{P}}^{(\mathsf{tm},\mathcal{F}')}(\mathsf{C}_{(s,\mathsf{ph})}) &= \mathsf{Pr}(\mathsf{ph}, \epsilon) \cdot \nu_{\mathcal{P}}(s) \\
&= \mathsf{Pr}(\mathsf{ph}, \epsilon) \cdot \textstyle\sum_{i=1}^{n_s} \nu_{\mathcal{P}}(s)\gamma_s(i) \\
&= \mathsf{Pr}(\mathsf{ph}, \epsilon) \cdot \textstyle\sum_{i=1}^{n_s} \nu_{\mathsf{ex}(\mathcal{P})}(s^{(i)}) \\
&= \mathsf{Pr}_{\mathsf{ex}(\mathcal{P})}^{(\mathsf{tm},\mathcal{F})}(\mathsf{ex}(\mathsf{C}_{(s,\mathsf{ph})})).
\end{aligned}$$

Now, let $\pi \in \mathsf{C}_\zeta)$, $|\zeta| = k$, $\pi{\downarrow_k} = \xi$ and $trace(\xi) = \sigma$. Furthermore, we assume that $\zeta = \zeta'\ E\ (s_k, \mathsf{ph}_k)$, $last(\zeta') = (s_{k-1}, \mathsf{ph}_{k-1})$. We distinguish the following cases:

1. If $(s_{k-1}, \mathsf{ph}_{k-1})$ is stable a race between the outgoing PH transitions of s_{k-1} and the action timers of the elements of the (finite) set

$$\{a_1, \ldots, a_m\} := \{a \in \mathsf{Act} \mid \mathsf{tm}_\sigma(a) = (\boldsymbol{\alpha}, T) \in \mathcal{R} \wedge s_{k-1} \xrightarrow{a}\ \}$$

takes place and is reflected by SPTP

$$\mathcal{Q} = P' \,||\, P_{a_1} \,||\, \ldots \,||\, P_{a_m}$$

as defined above (see page 143).

(a) Assume that $E \subseteq \mathbb{R}_{>0}$. For $\mathsf{ex}(\mathcal{P})$ we are interested in the measure of the set of all tm-paths in $\mathsf{ex}(\mathcal{P})$ that are of the form

$$(s_{k-1}^{(j_0)}, \mathsf{ph}_{k-1}) \xrightarrow{t_0} (s_{k-1}^{(j_1)}, \mathsf{ph}'_1) \xrightarrow{t_1} (s_{k-1}^{(j_2)}, \mathsf{ph}'_2) \xrightarrow{t_2} \ldots$$
$$\xrightarrow{t_{i-1}} (s_{k-1}^{(j_i)}, \mathsf{ph}'_i) \xrightarrow{t_i} (s_k^{(l)}, \mathsf{ph}_k)$$

where $i \geq 0$, $\sum_{h=0}^{i} t_h \in E$, $j_0, j_1 \ldots, j_i \in \{1, \ldots, n_{s_{k-1}}\}$, $1 \leq l \leq n_{s_k}$ and $\mathsf{ph}'_1, \mathsf{ph}'_2 \ldots, \mathsf{ph}'_i \in \mathcal{G}$. We start $\mathsf{ex}(\mathcal{P})$ with initial distribution $\boldsymbol{\gamma}_{s_{k-1}}$ and reach a representative of s_k before one of the SPTPs $\mathcal{P}_a$, $a \in \{a_1, \ldots, a_m\}$ reaches state 0 (recall that $\mathcal{P}_a$ is the component of $\mathcal{Q}$ that describes the behavior of the a-timer). Let Q' be the parallel composition of

$$P'' := (S_{\mathsf{ex}(\mathcal{P})}, \longrightarrow_{\mathsf{ex}(\mathcal{P})}, \emptyset, \boldsymbol{\gamma}_{s_{k-1}})$$

and the SPTPs $\mathcal{P}_{a_1}, \ldots, \mathcal{P}_{a_m}$. Furthermore, let $B', C' \subseteq S_{\mathcal{Q}'}$ with

$$\begin{aligned} B' &:= \{(u^{(j)} \,||\, u_{a_1} \,||\, \ldots \,||\, u_{a_m}) \mid j \in \{1, 2, \ldots, n_u\}, \\ &\qquad u = s_{k-1} \wedge u_{a_1}, \ldots, u_{a_m} > 0\} \\ C' &:= \{(u^{(j)} \,||\, u_{a_1} \,||\, \ldots \,||\, u_{a_m}) \mid j \in \{1, 2, \ldots, n_u\}, \\ &\qquad u = s_k \wedge u_{a_1}, \ldots, u_{a_m} > 0\} \end{aligned}.$$

Then

$$\begin{aligned} & \mathsf{Pr}_{\mathsf{ex}(\mathcal{P})}^{(\mathsf{tm},\mathcal{F})}(\mathsf{ex}(\mathsf{C}_\zeta)) \\ = \; & \mathsf{Pr}_{\mathsf{ex}(\mathcal{P})}^{(\mathsf{tm},\mathcal{F})}(\mathsf{ex}(\mathsf{C}_{\zeta'}) \cdot \mathsf{reach}^{\mathcal{Q}'}(B', C', E). \end{aligned}$$

But since P' and P'' have the same generator matrix (and so have $\mathcal{Q}$ and $\mathcal{Q}'$) it holds that

$$\mathsf{reach}^{\mathcal{Q}}(B, C, E) = \mathsf{reach}^{\mathcal{Q}'}(B', C', E)$$

and thus

$$\begin{aligned}
\mathsf{Pr}^{(\mathsf{tm},\mathcal{F})}_{\mathsf{ex}(\mathcal{P})}(\mathsf{ex}(\mathsf{C}_\zeta)) &= \mathsf{Pr}^{(\mathsf{tm},\mathcal{F})}_{\mathsf{ex}(\mathcal{P})}(\mathsf{ex}(\mathsf{C}_{\zeta'}) \cdot \mathsf{reach}^{\mathcal{Q}'}(B', C', E) \\
&= \mathsf{Pr}^{(\mathsf{tm},\mathcal{F})}_{\mathsf{ex}(\mathcal{P})}(\mathsf{ex}(\mathsf{C}_{\zeta'}) \cdot \mathsf{reach}^{\mathcal{Q}}(B, C, E) \\
&\overset{\text{ind. hyp.}}{=} \mathsf{Pr}^{(\mathsf{tm},\mathcal{F}')}_{\mathcal{P}}(\mathsf{C}_{\zeta'}) \cdot \mathsf{reach}^{\mathcal{Q}}(B, C, E) \\
&= \mathsf{Pr}^{(\mathsf{tm},\mathcal{F}')}_{\mathcal{P}}(\mathsf{C}_\zeta)
\end{aligned}$$

where B and C are defined as above.

(b) Assume that $E = \{a\} \times (x, y]$, $a \in \mathsf{Act}$, $x < y$. As stated above the probability to perform a within interval $(x, y]$ is composed of three factors (compare also page 144):

i. The probability that the a-timer expires earlier than all the other timers and before a PH transition of s_{k-1} can be taken (within $(x, y]$).

ii. The probability that after a is performed the new timer settings correspond to ph_k.

iii. The probability that the tm-scheduler decides for a certain a-transition.

For $\mathsf{ex}(\mathcal{P})$ we split up $\mathsf{Pr}^{(\mathsf{tm},\mathcal{F})}_{\mathsf{ex}(\mathcal{P})}(\mathsf{ex}(\mathsf{C}_\zeta))$ in the same way, i.e., into three factors according to i–iii. We prove that each factor is the same for $\mathsf{Pr}^{(\mathsf{tm},\mathcal{F}')}_{\mathcal{P}}(\mathsf{C}_\zeta)$ and $\mathsf{Pr}^{(\mathsf{tm},\mathcal{F})}_{\mathsf{ex}(\mathcal{P})}(\mathsf{ex}(\mathsf{C}_\zeta))$.

i. Consider the probability that the a-timer expires earlier than all the other timers and before a PH transition of s can be taken (within $(x, y]$). According to the construction of the modified PTP $\mathcal{Q}$ this probability equals $\mathsf{reach}^{\mathcal{Q}}(B, D, (x, y])$ where D is defined as above. For $\mathsf{ex}(\mathcal{P})$ we are interested in the measure of the set of all tm-paths of the form

$$(s_{k-1}^{(j_0)}, \mathsf{ph}_{k-1}) \xrightarrow{t_0} (s_{k-1}^{(j_1)}, \mathsf{ph}'_1) \xrightarrow{t_1} (s_{k-1}^{(j_2)}, \mathsf{ph}'_2) \xrightarrow{t_2} \dots$$
$$\xrightarrow{t_{i-1}} (s_{k-1}^{(j_i)}, \mathsf{ph}'_i) \xrightarrow{a,t_i} (s_k^{(l)}, \mathsf{ph}')$$

where $i \geq 0$, $x < \sum_{h=0}^{i} t_h \leq y$, $j_0, j_1 \dots, j_i \in \{1, \dots, n_{s_{k-1}}\}$, $1 \leq l \leq n_{s_k}$ and $\mathsf{ph}'_1, \mathsf{ph}'_2 \dots, \mathsf{ph}'_i, \mathsf{ph}' \in \mathcal{G}$ (note that for the first factor the timer initialization ph' is arbitrary). We use $\mathcal{Q}'$ to calculate the probability that corresponds to this set and define

$$D' := \big\{(u^{(j)} \,||\, u_{a_1} \,||\, \dots \,||\, u_{a_m}) \mid j \in \{1, 2, \dots, n_u\}$$
$$u = s_{k-1}, u_a = 0, u_b > 0, b \in \{a_1, \dots, a_m\} \setminus \{a\}\big\}.$$

It holds that $\mathsf{reach}^{\mathcal{Q}'}(B', D', (x, y])$ is the sought-after probability because the elements of D' correspond to the situation in which an a-transition of s_{k-1} is immediately enabled. A comparison with case 1b), factor i (see page 144) shows that

$$p'_1 := \mathsf{reach}^{\mathcal{Q}'}(B', D', (x, y]) = \mathsf{reach}^{\mathcal{Q}}(B, D, (x, y]) = p_1$$

where we use again that P' and P'' have the same generator matrix (and so have $\mathcal{Q}$ and $\mathcal{Q}'$).

ii. Let us consider the probability that after a is performed the new timer settings correspond to ph_k. Obviously, this factor equals $p_2 = \mathsf{Pr}(\mathsf{ph}_k, \sigma a) := p'_2$ for both, $\mathcal{P}$ and $\mathsf{ex}(\mathcal{P})$.

iii. For the third factor we make use of the definition of $\mathcal{F}$. Let $\hat{\xi} \in \mathsf{pathf}(\mathsf{tm}, \mathsf{ex}(\mathcal{P}))$ be such that $\mathsf{contr}(\hat{\xi}) = \xi'$ and let $\mathcal{F}'(\xi') =$

λ', $\mathcal{F}(\hat{\xi}) = \lambda$. Note that all tm-path fragments $\hat{\xi}$ that are of interest for $\mathsf{ex}(\mathsf{C})$ fulfill $\mathsf{contr}(\hat{\xi}) = \xi'$ or their maximal $\mathcal{P}$-observable prefix does and $\mathcal{F}$'s choice is the same for all such path fragments. We compute

$$\begin{aligned}
p_3 &= \sum\{\!|\lambda'(a,\mu')\cdot\mu'(s_k) \mid \exists\mu' : s_{k-1} \xrightarrow{a}\!\!\!\rightarrow \mu'|\!\}\\
&\overset{\text{Eq. A.8}}{=} \sum\{\!|\lambda(a,\mu)\cdot\mu'(s_k) \mid \exists\mu' : s_{k-1} \xrightarrow{a}\!\!\!\rightarrow \mu'\\
&\qquad \wedge\ \forall l : \mu'(s_k)\cdot\gamma_{s_k}(l) = \mu(s_k^{(l)})|\!\}\\
&= \sum\{\!|\lambda(a,\mu)\cdot\sum_{l=1}^{n_{s_k}}\gamma_{s_k}(l)\cdot\mu'(s_k) \mid \exists\mu' : s_{k-1} \xrightarrow{a}\!\!\!\rightarrow \mu'\\
&\qquad \wedge\ \forall l : \mu'(s_k)\cdot\gamma_{s_k}(l) = \mu(s_k^{(l)})|\!\}\\
&\overset{\text{Def. 4.2}}{=} \sum_{l=1}^{n_{s_k}}\sum\{\!|\lambda(a,\mu)\cdot\mu(s_k^{(l)}) \mid \exists\mu :\\
&\qquad \forall j\in\{1,\ldots,n_{s_{k-1}}\} : s_{k-1}^{(j)} \xrightarrow{a}\!\!\!\rightarrow \mu|\!\}\\
&:= p_3'.
\end{aligned}$$

We conclude that

$$\begin{aligned}
\mathsf{Pr}^{(\mathsf{tm},\mathcal{F})}_{\mathsf{ex}(\mathcal{P})}(\mathsf{ex}(\mathsf{C}_\zeta)) &= \mathsf{Pr}^{(\mathsf{tm},\mathcal{F})}_{\mathsf{ex}(\mathcal{P})}(\mathsf{ex}(\mathsf{C}_{\zeta'})\cdot p_1'\cdot p_2'\cdot p_3'\\
&\overset{\text{ind. hyp.}}{=} \mathsf{Pr}^{(\mathsf{tm},\mathcal{F}')}_{\mathcal{P}}(\mathsf{C}_{\zeta'})\cdot p_1'\cdot p_2'\cdot p_3'\\
&= \mathsf{Pr}^{(\mathsf{tm},\mathcal{F}')}_{\mathcal{P}}(\mathsf{C}_{\zeta'})\cdot p_1\cdot p_2\cdot p_3\\
&= \mathsf{Pr}^{(\mathsf{tm},\mathcal{F}')}_{\mathcal{P}}(\mathsf{C}_{\zeta}).
\end{aligned}$$

2. Now, let $(s_{k-1}, \mathsf{ph}_{k-1})$ be unstable. Then $E = \{(a,0)\}$, $a \in \mathsf{Act}_\tau$. This implies that $s_{k-1} \xrightarrow{a}\!\!\!\rightarrow$ and $\mathsf{ph}_{k-1}(a) = 0$ if $a \neq \tau$. Let p_2, p_2',p_3 and p_3' be defined as above except that $p_2 := 1$ and $p_2' := 1$ if $a = \tau$. With the

same arguments as for the case 1b), i and ii,

$$\begin{aligned}\mathsf{Pr}^{(\mathsf{tm},\mathcal{F})}_{\mathsf{ex}(\mathcal{P})}(\mathsf{ex}(\mathsf{C}_\zeta)) &= \mathsf{Pr}^{(\mathsf{tm},\mathcal{F})}_{\mathsf{ex}(\mathcal{P})}(\mathsf{ex}(\mathsf{C}_{\zeta'}) \cdot p_2' \cdot p_3' \\ &= \mathsf{Pr}^{(\mathsf{tm},\mathcal{F}')}_{\mathcal{P}}(\mathsf{C}_{\zeta'}) \cdot p_2 \cdot p_3 \\ &= \mathsf{Pr}^{(\mathsf{tm},\mathcal{F}')}_{\mathcal{P}}(\mathsf{C}_\zeta)\end{aligned}$$

which completes the whole proof.

□

A.3 Proof of Theorem 7.1

Theorem A.2

$=_{\mathsf{te}}$ *is finer than* $=_{\mathsf{de}}$.

Proof. Let us now prove that $\mathcal{P}_1 =_{\mathsf{te}} \mathcal{P}_2$ implies $\mathcal{P}_1 =_{\mathsf{de}} \mathcal{P}_2$. The idea is to construct a test process $\mathcal{T}_{\mathsf{tm}}$ which simulates the enviroment conditions described by the timer settings tm. The proof consists of five steps. In the first step, we extend $\mathcal{P}_1$ and $\mathcal{P}_2$ by action transitions with labels $z_a, a \in \mathsf{Act}$ which are used to communicate to $\mathcal{T}_{\mathsf{tm}}$ that $\mathcal{P}_1$ performed action a. This means that timer settings change from tm_σ to $\mathsf{tm}_{\sigma a}$ and $\mathcal{T}_{\mathsf{tm}}$ has to give input stimuli according to $\mathsf{tm}_{\sigma a}$. The modified processes are called $\mathcal{P}_1'$ and $\mathcal{P}_2'$. The second step is concerned with the construction of the test process $\mathcal{T}_{\mathsf{tm}}$. In the third step an HR-scheduler for $\mathcal{P}_1'|\mathcal{T}_{\mathsf{tm}}$ is defined which simulates the decisions of tm-scheduler $\mathcal{F}_1$ for $\mathcal{P}_1$. Then, in step four, we prove by induction on the length of a cylinder set of paths that $\mathcal{P}_1'|\mathcal{T}_{\mathsf{tm}}$ under $\mathcal{D}$ produces trace observations with the same probability as $\mathcal{P}_1$ under $\mathcal{F}_1$ does. Finally (step five) we show $\mathcal{P}_1 =_{\mathsf{te}} \mathcal{P}_2$ implies $\mathcal{P}_1 =_{\mathsf{de}} \mathcal{P}_2$ by adding success actions to $\mathcal{T}_{\mathsf{tm}}$.

1. Assume that $\mathcal{P}_1$ and $\mathcal{P}_2$ are SPTPs. For $i \in \{1,2\}$, we construct a modified version $\mathcal{P}_i'$ of $\mathcal{P}_i$ as follows: Each transition $s \xrightarrow{a} \mu$, $a \in \mathsf{Act}$ of

$\mathcal{P}_i$ is replaced by transitions $s \xrightarrow{a}\!\!\!\!\rightarrow \delta_u$ and $u \xrightarrow{z_a}\!\!\!\!\rightarrow \mu$ where $u = u(s, a, \mu)$ is a fresh state and z_a is a fresh action. Thus, the state space is extended by the set $\{\!|u \mid u = u(s, a, \mu), s \xrightarrow{a}\!\!\!\!\rightarrow \mu, a \neq \tau|\!\}$ and the set of actions by $\{z_a, a \in \mathsf{Act}\}$.

Obviously,

$$\mathcal{P}_1 =_{\mathsf{te}} \mathcal{P}_2 \iff \mathcal{P}_1' =_{\mathsf{te}} \mathcal{P}_2'$$

and

$$\mathcal{P}_1 =_{\mathsf{de}} \mathcal{P}_2 \iff \mathcal{P}_1' =_{\mathsf{de}} \mathcal{P}_2'.$$

2. Suppose that $\mathsf{tm} : \mathsf{Act}^* \times \mathsf{Act} \rightarrow \big(\mathcal{R} \cup \{\mathsf{I}, \mathsf{B}\}\big)$ represents the timer settings for the delayed trace machine. We construct a test process $\mathcal{T}_{\mathsf{tm}}$ as the union of SPTPs T_σ, $\sigma \in \mathsf{Act}^*$ in the following way. Firstly, for each σ process T_σ is given by

$$\mathcal{T}_\sigma := \mathcal{T}_\sigma^{a_1} \,||\, \mathcal{T}_\sigma^{a_2} \,||\, \ldots \,||\, \mathcal{T}_\sigma^{a_n}$$

where

$$\{a_1, a_2, \ldots, a_n\} = \{a \in \mathsf{Act} \mid \mathsf{tm}_\sigma(a) \in \mathcal{R} \cup \{\mathsf{I}\}\} =: \mathsf{Act}(\mathsf{tm}_\sigma) \neq \emptyset.$$

Recall that we impose the constraint that $\mathsf{Act}(\mathsf{tm}_\sigma)$ is finite. If $\mathsf{Act}(\mathsf{tm}_\sigma) \neq \emptyset$ then $\mathcal{T}_\sigma$ consists of a single absorbing state without any outgoing transitions.

If $\mathsf{tm}_\sigma(a) = (\boldsymbol{\alpha}, T)$ then SPTP $\mathcal{T}_\sigma^a = (S_\sigma^a, \longrightarrow_\sigma^a, \longrightarrow\!\!\!\!\rightarrow_\sigma^a, \nu_\sigma^a)$ is defined as follows:[1]

- $S_\sigma^a := \{i \mid -1 \leq i \leq k\}$ where k is the order of $(\boldsymbol{\alpha}, T)$,
- $\longrightarrow_\sigma^a := \{(i, T_{ij}, j) \mid i, j \geq 1, i \neq j\} \cup \{(i, \mathbf{T}_i^0, 0) \mid i, j \geq 1, i \neq j\}$,

[1] We briefly write $(s, (\boldsymbol{\alpha}, T), s')$ instead of $(s, (\boldsymbol{\alpha}, T), \delta_{s'})$ and (s, a, s') instead of $(s, a, \delta_{s'})$.

- $\longrightarrow\!\!\!\!\!\rightarrow^{a}_{\sigma} := \{(0, a, -1)\}$,
- ν^a_σ is such that $\nu^a_\sigma(i) = \alpha(i)$ if $1 \le i \le k$ and $\nu^a_\sigma(i) = 0$ otherwise.

Intuitively, $\mathcal{T}^a_\sigma$ performs action a after a PH delay with representation $(\boldsymbol{\alpha}, T)$. If $\mathsf{tm}_\sigma(a) = \mathsf{I}$ then SPTP $\mathcal{T}^a_\sigma = (S^a_\sigma, \longrightarrow^{a}_{\sigma}, \longrightarrow\!\!\!\!\!\rightarrow^{a}_{\sigma}, \nu^a_\sigma)$ performs a immediately, i.e. $S^a_\sigma = \{0, -1\}$, $\longrightarrow^{a}_{\sigma} = \{\}$, $\longrightarrow\!\!\!\!\!\rightarrow^{a}_{\sigma} = \{(0, a, -1)\}$, $\nu^a_\sigma(0) = 1$ and $\nu^a_\sigma(-1) = 0$.

We add superscript a to indicate the elements of S^a_σ. We write, for example, i^a instead of i for $i \in S^a_\sigma$. Furthermore, let the elements of

$$\mathcal{T}_\sigma = \mathcal{T}^{a_1}_\sigma \,||\, \mathcal{T}^{a_2}_\sigma \,||\, \ldots \,||\, \mathcal{T}^{a_n}_\sigma$$

be given by $(S_\sigma, \longrightarrow_\sigma, \longrightarrow\!\!\!\!\!\rightarrow_\sigma, \nu_\sigma)$ and elements of S_σ have the form $s = (i^{a_1} \,||\, i^{a_2} \,||\, \ldots \,||\, i^{a_n})$ if $\mathsf{Act}(\mathsf{tm}_\sigma) = \{a_1, a_2, \ldots, a_n\}$. In order to achieve $\cap_{\sigma \in \mathsf{Act}^*} S_\sigma = \emptyset$, we rename the states of $\mathcal{T}_\sigma$ as follows: let $(\sigma, i^{a_1} \,||\, i^{a_2} \,||\, \ldots \,||\, i^{a_n})$ be state $(i^{a_1} \,||\, i^{a_2} \,||\, \ldots \,||\, i^{a_n}) \in S_\sigma$.

Now, $\mathcal{T}_{\mathsf{tm}} := (S_{\mathsf{tm}}, \longrightarrow_{\mathsf{tm}}, \longrightarrow\!\!\!\!\!\rightarrow_{\mathsf{tm}}, \nu_{\mathsf{tm}})$ is constructed from the $\mathcal{T}_\sigma$'s as follows:

- $S_{\mathsf{tm}} := \cup_{\sigma \in \mathsf{Act}^*} S_\sigma$,
- $\longrightarrow_{\mathsf{tm}} := \cup_{\sigma \in \mathsf{Act}^*} \longrightarrow_\sigma$,
- $\longrightarrow\!\!\!\!\!\rightarrow_{\mathsf{tm}} := (\cup_{\sigma \in \mathsf{Act}^*} \longrightarrow\!\!\!\!\!\rightarrow_\sigma) \cup Z$ where[2]

$$\begin{aligned} Z \;:=\; & \{(s, z_a, \nu_{\sigma a}) \mid s = (\sigma, i^{a_1} \,||\, i^{a_2} \,||\, \ldots \,||\, i^{a_n}) \in S_\sigma, \\ & \exists l : a = a_l, i^a = -1\}, \end{aligned}$$

- $\nu_{\mathsf{tm}} := \nu_\epsilon$.

[2]Recall that for a set A we extend distribution $\alpha \in \mathsf{dis}(A)$ on set $B \supset A$ by letting $\alpha(s) = 0$ if $s \in B \setminus A$.

This means that in $\mathcal{T}_{\mathsf{tm}}$ for all σ and all a process $\mathcal{T}_\sigma$ is connected to $\mathcal{T}_{\sigma a}$ via a z_a-transition.

Obviously, for each $\xi \in \mathsf{pathf}(\mathcal{T}_{\mathsf{tm}})$ with $last(\xi) = (\sigma, i^{a_1} \,||\, i^{a_2} \,||\, \ldots \,||\, i^{a_n})$ and

$$trace(\xi) = b_1 z_{b_1} b_2 z_{b_2} \ldots b_m z_{b_m} \text{ or} \tag{A.9}$$

$$trace(\xi) = b_1 z_{b_1} b_2 z_{b_2} \ldots b_{m-1} z_{b_{m-1}} b_m \tag{A.10}$$

it holds that $\mathsf{Act}(\mathsf{tm}_\sigma) = \{a_1, a_2, \ldots, a_n\}$ and $\sigma = b_1 b_2 \ldots b_m$ in the case of A.9 and $\sigma = b_1 b_2 \ldots b_{m-1}$ in the case of A.10. Moreover, all path fragments of $\mathcal{P}_1' | \mathcal{T}_{\mathsf{tm}}$ containing at least one visible action have traces of the form A.9 or A.10.

3. The next step is to connect path fragments of $\mathcal{P}_1' | \mathcal{T}_{\mathsf{tm}}$ and tm-path fragments of $\mathcal{P}_1$ in order to define a scheduler for $\mathcal{P}_1' | \mathcal{T}_{\mathsf{tm}}$. The time until an action, say, a is enabled in $\mathcal{T}_{\mathsf{tm}}$ is represented by k single phased PH transitions if the order of $\mathsf{tm}_\sigma(a)$ is k. In contrast to that a tm-path of $\mathcal{P}_1$ captures this time delay in one step and hides the phase changes of the action timer. Moreover, action transitions are not followed by a z-transitions as in $\mathcal{P}_1' | \mathcal{T}_{\mathsf{tm}}$. We modify the elements of the set $\mathsf{pathf}(\mathcal{P}_1' | \mathcal{T}_{\mathsf{tm}})$ in the following way:

   Let $\xi \in \mathsf{pathf}(\mathcal{P}_1' | \mathcal{T}_{\mathsf{tm}})$.

   - Whenever $s \xrightarrow{a}\!\!\!\!\to s' \xrightarrow{z_a}\!\!\!\!\to s''$ is a part of ξ for some $a \in \mathsf{Act}$ these two transitions are replaced by $s \xrightarrow{a}\!\!\!\!\to s''$. Note that this removes z_a completely from all path fragments since it does not occur in any other way.

- Whenever ξ contains a maximal subsequence of the form

$$\begin{array}{ll} v|(\sigma, i^{a_1} \,||\, i^{a_2} \,||\, \ldots \,||\, i^{a_n}) & \xrightarrow{t_1} \\ v|(\sigma, j^{a_1} \,||\, j^{a_2} \,||\, \ldots \,||\, j^{a_n}) & \xrightarrow{t_2} \ldots \xrightarrow{t_N} \\ v|(\sigma, l^{a_1} \,||\, l^{a_2} \,||\, \ldots \,||\, l^{a_n}) & \xrightarrow{a}\!\!\!\!\rightarrow \\ v'|(\sigma, k^{b_1} \,||\, k^{b_2} \,||\, \ldots \,||\, k^{b_m}) & \end{array} \tag{A.11}$$

i.e. $\mathcal{P}'_1$ remains in its current state whereas $\mathcal{T}_{\mathsf{tm}}$ performs PH transitions until action a can be carried out synchronously, we replace this subsequence by

$$v|(\sigma, i^{a_1} \,||\, i^{a_2} \,||\, \ldots \,||\, i^{a_n}) \xrightarrow{a,t} v'|(\sigma, k^{b_1} \,||\, k^{b_2} \,||\, \ldots \,||\, k^{b_m})$$

where $t = \sum_{h=1}^{N} t_h$. Note that an HR-scheduler $\mathcal{D}$ for $\mathcal{P}'_1|\mathcal{T}_{\mathsf{tm}}$ has nothing to decide in intermediate stable states since no action transition is possible in these states (but $\mathcal{D}$ has to decide which of the a-transitions of v is taken).

- Whenever ξ contains a maximal subsequence of the form

$$\begin{array}{ll} v|(\sigma, i^{a_1} \,||\, i^{a_2} \,||\, \ldots \,||\, i^{a_n}) & \xrightarrow{t_1} \\ v|(\sigma, j^{a_1} \,||\, j^{a_2} \,||\, \ldots \,||\, j^{a_n}) & \xrightarrow{t_2} \ldots \xrightarrow{t_{N-1}} \\ v|(\sigma, l^{a_1} \,||\, l^{a_2} \,||\, \ldots \,||\, l^{a_n}) & \xrightarrow{t_N} \\ v'|(\sigma, l^{a_1} \,||\, l^{a_2} \,||\, \ldots \,||\, l^{a_n}) & \end{array} \tag{A.12}$$

i.e. $\mathcal{P}'_1$ remains in its current state whereas $\mathcal{T}_{\mathsf{tm}}$ performs PH transitions and finally $\mathcal{P}'_1$ executes a PH transition, we replace this subsequence by

$$v|(\sigma, i^{a_1} \,||\, i^{a_2} \,||\, \ldots \,||\, i^{a_n}) \xrightarrow{t} v'|(\sigma, l^{a_1} \,||\, l^{a_2} \,||\, \ldots \,||\, l^{a_n})$$

where $t = \sum_{h=1}^{N} t_h$. Again an HR-scheduler $\mathcal{D}$ for $\mathcal{P}'_1|\mathcal{T}_{\mathsf{tm}}$ has nothing to decide in the intermediate stable states.

Let $\widetilde{\mathsf{contr}}(\xi)$ denote path fragment ξ after the three modifications and let $\widetilde{\mathsf{pathf}}(\mathcal{P}_1'|\mathcal{T}_{\mathsf{tm}})$ denote the set of all such path fragments $\widetilde{\mathsf{contr}}(\xi)$.

The partial function $f : \widetilde{\mathsf{pathf}}(\mathcal{P}_1'|\mathcal{T}_{\mathsf{tm}}) \to \mathsf{pathf}(\mathcal{P}_1, \mathsf{tm})$ is defined inductively by

$$f(v|(\epsilon, i^{a_1} \,||\, i^{a_2} \,||\, \ldots \,||\, i^{a_n})) = (v, \mathsf{ph})$$

where for all $a \in \mathsf{Act}(\mathsf{tm}_\epsilon) = \{a_1, a_2, \ldots, a_n\}$ we have $\mathsf{ph}(a) = i^a \geq 0$ (from the definition of the initial distributions ν_ϵ^a we have that initially $i^a \geq 0$). Furthermore, if $\widetilde{\mathsf{contr}}(\xi), \widetilde{\mathsf{contr}}(\xi') \in \widetilde{\mathsf{pathf}}(\mathcal{P}_1'|\mathcal{T}_{\mathsf{tm}})$ and

$$\widetilde{\mathsf{contr}}(\xi) = \widetilde{\mathsf{contr}}(\xi') \xrightarrow{\mathsf{e}} v|(\sigma, i^{a_1} \,||\, i^{a_2} \,||\, \ldots \,||\, i^{a_n})$$

we define

$$f(\widetilde{\mathsf{contr}}(\xi)) = f(\widetilde{\mathsf{contr}}(\xi')) \xrightarrow{\hat{\mathsf{e}}} (v, \mathsf{ph})$$

where again for all $a \in \{a_1, a_2, \ldots, a_n\}$ we have $\mathsf{ph}(a) = i^a \geq 0$ and $\hat{\mathsf{e}} = \mathsf{e}$ if $\mathsf{e} = (a, t)$ or $\mathsf{e} = t$, $\hat{\mathsf{e}} = (\mathsf{e}, 0)$ if $\mathsf{e} \in \mathsf{Act}_\tau$. $f(\widetilde{\mathsf{contr}}(\xi))$ is undefined whenever the states i^a or event e do not fulfill the above conditions or if $f(\widetilde{\mathsf{contr}}(\xi'))$ is undefined.

Now, we are able to define HR-scheduler $\mathcal{D}$ for $\mathcal{P}_1'|\mathcal{T}_{\mathsf{tm}}$ if a tm-scheduler $\mathcal{F}_1$ for $\mathcal{P}_1$ is given. Let ξ be a path fragment of $\mathcal{P}_1'|\mathcal{T}_{\mathsf{tm}}$, $last(\xi) = s = v|(\sigma, i^{a_1} \,||\, i^{a_2} \,||\, \ldots \,||\, i^{a_n})$ and $trace(\widetilde{\mathsf{contr}}(\xi)) = \sigma$. Then $\mathcal{D}(\xi) = \lambda$ is such that

- if there exists a with $i^a = -1$ and $s \xrightarrow{z_a}\!\!\!\!\rightarrow \mu$ then $\lambda(z_a, \mu) = 1$,
- if $s \not\xrightarrow{\tau}\!\!\!\!\rightarrow$ and $i^a > 0$ for all $a \in \{a_1, a_2, \ldots, a_n\}$, $v \xrightarrow{a}\!\!\!\!\rightarrow$ then $\lambda^\perp = 1$,
- if $i^{a_k} \geq 0$ for $1 \leq k \leq n$ and $\{b \in \mathsf{Act}_\tau \mid v \xrightarrow{b}\!\!\!\!\rightarrow, i^b = 0\} = \{b_1, b_2, \ldots, b_m\} \neq \emptyset$ then $f(\widetilde{\mathsf{contr}}(\xi)) \in \mathsf{pathf}(\mathcal{P}_1, \mathsf{tm})$ and λ is defined as follows: Let $\lambda' := \mathcal{F}_1(f(\widetilde{\mathsf{contr}}(\xi)))$. Transition

$$s \xrightarrow{b_k}\!\!\!\!\rightarrow u|(\sigma, j^{a_1} \,||\, j^{a_2} \,||\, \ldots \,||\, j^{a_n}) = s', \quad b_k \neq \tau, 1 \leq k \leq m,$$

is chosen with probability $\lambda(b_k, s') := \lambda'(b_k, \mu)$ if $u = u(v, b_k, \mu)$ and

$$j^{a_l} = \begin{cases} i^{a_l} & \text{if } a_l \neq b_k \\ -1 & \text{otherwise,} \end{cases}$$

for all $l \in \{1, \ldots, n\}$. However, if $b_k = \tau$ then transition $s \xrightarrow{\tau} \mu$, is chosen with probability $\lambda(\tau, \mu) := \lambda'(\tau, \mu')$ if

$$\mu'(v') = \mu(v'|(\sigma, i^{a_1} \,||\, i^{a_2} \,||\, \ldots \,||\, i^{a_n}))$$

for all v'. All remaining transitions have probability zero.

By using the fact that $\mathcal{F}_1$ is a tm-scheduler, it can be shown that $\mathcal{D} \in \mathsf{HR}(\mathcal{P}_1'|\mathcal{T}_{\mathsf{tm}})$.

4. The next step is concerned with the proof of the fact that for all $\sigma = a_1 a_2, \ldots, a_n \in \mathsf{Act}^*$, $\sigma' = a_1 z_{a_1} a_2 z_{a_2} \ldots a_n z_{a_n}$ we have

$$\begin{aligned} & \mathsf{Pr}^{\mathcal{D}}(\{\pi \in \mathsf{path}(\mathcal{P}_1'|\mathcal{T}_{\mathsf{tm}}) \mid \exists i : trace(\pi{\downarrow_i}) = \sigma', time(\pi{\downarrow_i}) \leq t\}) \\ = \; & \mathsf{Pr}^{(\mathsf{tm},\mathcal{F}_1)}(\{\pi \in \mathsf{path}(\mathcal{P}_1, \mathsf{tm}) \mid \exists i : trace(\pi{\downarrow_i}) = \sigma, time(\pi{\downarrow_i}) \leq t\}. \end{aligned}$$

This is shown by going back to the cylinder set construction for probability measures. Let C_ζ be a cylinder set of tm-path in $\mathcal{P}_1$ and let $\Xi_\zeta \subseteq \mathsf{pathf}(\mathcal{P}_1, \mathsf{tm})$ be the set of tm-path fragments (as defined above). We set

$$\widetilde{\mathsf{ex}}(\mathsf{C}_\zeta) := \{\pi \in \mathsf{path}(\mathcal{P}_1'|\mathcal{T}_{\mathsf{tm}}) \mid \exists j : f(\widetilde{\mathsf{contr}}(\pi{\downarrow_j})) \in \Xi_\zeta\}.$$

We prove by induction on the length k of ζ that for all such cylinder sets C_ζ

$$\mathsf{Pr}^{\mathcal{D}}\Big(\widetilde{\mathsf{ex}}(\mathsf{C}_\zeta)\Big) = \mathsf{Pr}^{(\mathsf{tm},\mathcal{F}_1)}\Big(\mathsf{C}_\zeta\Big).$$

Let $\mathsf{tm}_\epsilon = \{a_1, a_2, \ldots, a_n\}$ and $\zeta = (v, \mathsf{ph})$. For $k = 1$ we get

$$\begin{aligned}
\mathsf{Pr}^{(\mathsf{tm},\mathcal{F}_1)}\Big(\mathsf{C}_{(v,\mathsf{ph})}\Big) &= \nu_{\mathcal{P}_1}(v) \cdot \textstyle\prod_{a:\mathsf{tm}_\epsilon(a)=(\boldsymbol{\alpha},T)\in\mathcal{R}} \alpha(\mathsf{ph}(a)) \\
&= \nu_{\mathcal{P}'_1}(v) \cdot \textstyle\prod_{j=1}^{n} \nu_\epsilon^{a_j}(\mathsf{ph}(a_j)) \\
&= \nu_{\mathcal{P}'_1}(v) \cdot \nu_{\mathsf{tm}}(\epsilon, \mathsf{ph}(a_1) \,||\, \mathsf{ph}(a_2) \,||\, \ldots \,||\, \mathsf{ph}(a_n)) \\
&= \mathsf{Pr}^{\mathcal{D}}\Big(\widetilde{\mathsf{ex}}(\mathsf{C}_{(v,\mathsf{ph})})\Big)
\end{aligned}$$

where the product equals 1 if there exists no action $a \in \mathsf{Act}(\mathsf{tm}_\epsilon)$ with $\mathsf{tm}_\epsilon(a) \in \mathcal{R}$. If now $k > 0$, $trace(\zeta) = \sigma$ and

$$\begin{aligned}
\zeta &= (s_1, \mathsf{ph}_1)\ E_1\ (s_2, \mathsf{ph}_2)\ E_2 \ldots E_{k-1}\ (s_k, \mathsf{ph}_k), \\
\zeta' &:= (s_1, \mathsf{ph}_1)\ E_1\ (s_2, \mathsf{ph}_2)\ E_2 \ldots E_{k-2}\ (s_{k-1}, \mathsf{ph}_{k-1}),
\end{aligned}$$

we distinguish three cases:

(a) If $E_{k-1} = \{a\} \times (x, y], x < y$ (and therefore $t \in J_{k-1}$, $s \not\twoheadrightarrow$) we recall modification A.11 and let $\mathsf{Act}(\mathsf{tm}_\sigma) =: \{a_1, a_2, \ldots, a_n\}$,

$$\begin{aligned}
v &:= s_{k-1}|(\sigma, \mathsf{ph}_{k-1}(a_1) \,||\, \mathsf{ph}_{k-1}(a_2) \,||\, \ldots \,||\, \mathsf{ph}_{k-1}(a_n)). \\
v' &:= s_k|(\sigma, \mathsf{ph}_k(a_1) \,||\, \mathsf{ph}_k(a_2) \,||\, \ldots \,||\, \mathsf{ph}_k(a_n)).
\end{aligned}$$

Furthermore, we define $A_{>0}$ as the set of all states of the form

$$s_{k-1}|(\sigma, k^{a_1} \,||\, k^{a_2} \,||\, \ldots \,||\, k^{a_n})$$

such that $k^b > 0$ for all $b \in \{a_1, \ldots, a_n\}, v \xrightarrow{b}\!\!\!\!\twoheadrightarrow$ and A_a is the set of all such states in which $\mathcal{T}_\sigma^a$ is in state 0, i.e. $k^a = 0$. Then

$$\mathsf{Pr}^{\mathcal{D}}\Big(\widetilde{\mathsf{ex}}(\mathsf{C}_\zeta)\Big) = \mathsf{Pr}^{\mathcal{D}}\Big(\widetilde{\mathsf{ex}}(\mathsf{C}_{\zeta'})\Big) \cdot p'_1 \cdot p'_2 \cdot p'_3$$

where

$$p'_1 = \mathsf{reach}_v^{\mathcal{P}'_1|\mathcal{T}_\sigma}(A_{>0}, A_a, (x, y]).$$

The probability that $\mathcal{T}_{\mathsf{tm}}$ changes its state according to ph_k is given by

$$p'_2 = \nu_{\sigma a}(\sigma, \mathsf{ph}_k(a_1) \,||\, \mathsf{ph}_k(a_2) \,||\, \ldots \,||\, \mathsf{ph}_k(a_n)).$$

which equals p_2 (see Equation 6.4).

The probability p'_3 that $\mathcal{D}$ decides for an a-transition in v such that $\mathcal{P}'_1$ reaches state $u(s_{k-1}, a, \mu)$ multiplied by $\mu(s_k)$ does, by contruction of $\mathcal{D}$, only depend on the choice of $\mathcal{F}_1(\zeta)$ and equals

$$p'_3 = \sum \left\{ \left| \mathcal{F}_1(\zeta)(a, \mu) \cdot \mu(s_k) \mid \exists \mu : s_{k-1} \xrightarrow{a} \mu \right| \right\} = p_3.$$

Note that this includes the execution of the z_a-action. Now, compare the definition of p_1 in Equation 6.6 on page 144 for the construction of the measure $\mathsf{Pr}^{(\mathsf{tm}, \mathcal{F}_1)}$ (case 2(a)). It holds that equals $p'_1 = p_1$, i.e. the probability that the a-timer expires earlier than all other timers and before a PH transition of s_{k-1} can be taken (within $(x, y]$). In both cases we consider the race between the PH transitions of s_{k-1} and PH transitions labeled by

$$\{(\delta_i, T) \mid s_{k-1} \xrightarrow{a'}, \mathsf{tm}_\sigma(a') = (\boldsymbol{\alpha}, T), \mathsf{ph}(a') = i\}.$$

For $\mathsf{Pr}^{(\mathsf{tm}, \mathcal{F}_1)}$ the probability that the transition associated to $\mathsf{tm}_\sigma(a)$ wins this race within $(x, y]$ is calculated by the construction of $\mathcal{Q}'$ on page 144. The part of the generator matrix of $\mathcal{Q}'$ which expresses the race in state s_{k-1} equals the part of $\mathcal{P}'_1|\mathcal{T}_\sigma$'s generator for set $A_{>0} \cup A_a$.

Using the induction hypothesis we combine

$$\begin{aligned}
\mathsf{Pr}^{\mathcal{D}}\Big(\widetilde{\mathsf{ex}}(\mathsf{C}_\zeta)\Big) &= \mathsf{Pr}^{\mathcal{D}}\Big(\widetilde{\mathsf{ex}}(\mathsf{C}_\zeta)\Big) \cdot p1' \cdot p'_2 \cdot p'_3 \\
&= \mathsf{Pr}^{(\mathsf{tm}, \mathcal{F}_1)}\Big(\mathsf{C}_{\zeta'}\Big) \cdot p_1 \cdot p_2 \cdot p_3 \\
&= \mathsf{Pr}^{(\mathsf{tm}, \mathcal{F}_1)}\Big(\mathsf{C}_\zeta\Big).
\end{aligned}$$

(b) Assume that $E_{k-1} = (x, y]$,$0 \leq x < y$. From modification A.12 (see page 194) we know that $\mathcal{P}_1'$ performed a PH transition from s_{k-1} to s_k while $\mathcal{T}_{\mathsf{tm}}$ reached state $(\sigma, j^{b_1} \,||\, j^{b_2} \,||\, \ldots \,||\, j^{b_m})$ from state $(\sigma, i^{a_1} \,||\, i^{a_2} \,||\, \ldots \,||\, i^{a_n})$ within $\mathcal{T}_\sigma$ where $\mathsf{ph}_{k-1}(b_l) = j^{b_l}$, $1 \leq l \leq m$, $\mathsf{ph}_k(a_h) = i^{a_h}$, $1 \leq h \leq n$. This means that

$$\mathsf{Act}(\mathsf{tm}_\sigma) = \{b_1, b_2, \ldots, b_m\} = \{a_1, a_2, \ldots, a_n\}.$$

Hence, without loss of generality we write $(\sigma, j^{a_1} \,||\, j^{a_2} \,||\, \ldots \,||\, j^{a_n})$ for $(\sigma, j^{b_1} \,||\, j^{b_2} \,||\, \ldots \,||\, j^{b_m})$. Let

$$\begin{aligned} v &:= s_{k-1}|(\sigma, i^{a_1} \,||\, i^{a_2} \,||\, \ldots \,||\, i^{a_n}). \\ v' &:= s_{k-1}|(\sigma, j^{a_1} \,||\, j^{a_2} \,||\, \ldots \,||\, j^{a_n}). \end{aligned}$$

Then

$$\mathsf{Pr}^{\mathcal{D}}\Big(\widetilde{\mathsf{ex}}(\mathsf{C}_\zeta)\Big) = \mathsf{Pr}^{\mathcal{D}}\Big(\widetilde{\mathsf{ex}}(\mathsf{C}_{\zeta'})\Big) \cdot \mathsf{reach}_v^{\mathcal{P}_1|\mathcal{T}_{\mathsf{tm}}}(A, \{v'\}, E_{k-1})$$

where

$$A = \Big\{\big(s_{k-1}|(\sigma, l^{a_1} \,||\, l^{a_2} \,||\, \ldots \,||\, l^{a_n})\big) \mid \forall a \in \{a_1, a_2, \ldots, a_n\} : l^a \leq 0 \implies s_{k-1} \overset{a}{\not\twoheadrightarrow} \Big\}.$$

But by construction this equals the probability $\mathsf{reach}^{\mathcal{Q}}(B, C, (x, y])$ in Equation 6.5 on page 143. Therefore, our statement also holds in this case, i.e.

$$\begin{aligned} \mathsf{Pr}^{\mathcal{D}}\Big(\widetilde{\mathsf{ex}}(\mathsf{C}_\zeta)\Big) &= \mathsf{Pr}^{\mathcal{D}}\Big(\widetilde{\mathsf{ex}}(\mathsf{C}_\zeta)\Big) \cdot \mathsf{reach}_v^{\mathcal{P}_1|\mathcal{T}_{\mathsf{tm}}}(A, \{v'\}, E_{k-1}) \\ &= \mathsf{Pr}^{(\mathsf{tm}, \mathcal{F}_1)}\Big(\mathsf{C}_{\zeta'}\Big) \cdot \mathsf{reach}^{\mathcal{Q}}(B, C, (x, y]) \\ &= \mathsf{Pr}^{(\mathsf{tm}, \mathcal{F}_1)}\Big(\mathsf{C}_\zeta\Big). \end{aligned}$$

(c) Finally, if If $E_{k-1} = \{(a, 0\}$, $a \in \mathsf{Act}_\tau$ the situation is similar to the first case. If $a \neq \tau$, the probability p_2 of timer initialization according to ph_k equals

$$p_2' = \nu_{\sigma a}(\sigma, \mathsf{ph}_k(b_1) \,||\, \mathsf{ph}_k(b_2) \,||\, \ldots \,||\, \mathsf{ph}_k(b_m)).$$

where $\mathsf{Act}(\mathsf{tm}_{\sigma a}) =: \{b_1, b_2, \ldots, b_m\}$, and the branching probability p_3' according to -scheduler $\mathcal{D}$ equals

$$p_3 := \sum \Big\{\Big| \mathcal{F}_1(\zeta)(a, \mu) \cdot \mu(s_k) \mid \exists \mu : s_{k-1} \xrightarrow{a} \mu \Big|\Big\}. \tag{A.13}$$

If $a = \tau$ the p_2'-term (p_2-term) drops out and the statement follows directly.

Hence, for all cylinder sets C_ζ

$$\mathsf{Pr}^{\mathcal{D}}\Big(\widetilde{\mathsf{ex}}(\mathsf{C}_\zeta)\Big) = \mathsf{Pr}^{(\mathsf{tm},\mathcal{F}_1)}\Big(\mathsf{C}_\zeta\Big).$$

But then for all $\sigma = a_1 a_2, \ldots, a_n \in \mathsf{Act}^*$, $\sigma' = a_1 z_{a_1} a_2 z_{a_2} \ldots a_n z_{a_n}$ it holds that

$$\begin{aligned} &\mathsf{tr}^{\mathcal{D}}_{\mathcal{P}_1'|\mathcal{T}_{\mathsf{tm}}}(\sigma', t) \\ =& \mathsf{Pr}^{\mathcal{D}}(\{\pi \in \mathsf{path}(\mathcal{P}_1'|\mathcal{T}_{\mathsf{tm}}) \mid \exists i : trace(\pi{\downarrow_i}) = \sigma', time(\pi{\downarrow_i}) \leq t\}) \\ =& \mathsf{Pr}^{(\mathsf{tm},\mathcal{F}_1)}(\{\pi \in \mathsf{path}(\mathcal{P}_1, \mathsf{tm}) \mid \exists i : trace(\pi{\downarrow_i}) = \sigma, time(\pi{\downarrow_i}) \leq t\} \\ =& \mathsf{de}^{(\mathsf{tm},\mathcal{F}_1)}_{\mathcal{P}_1}(\sigma, t). \end{aligned} \tag{A.14}$$

5. In the final step we incorporate success actions θ_σ in $\mathcal{T}_{\mathsf{tm}}$. Each transition $s_\sigma \xrightarrow{z_a} \nu_{\sigma a}, s_\sigma \in S_\sigma$ is replaced by $s_\sigma \xrightarrow{z_a} \mu_{\sigma a}$ with $\mu_{\sigma a}(s) = 0.5 \cdot \nu_{\sigma a}(s)$ for all $s \in S_{\sigma a}$ and $\mu_{\sigma a}(w_{\sigma a}) = 0.5$. Here, $w_{\sigma a}$ are fresh states having a single transition which is a loop labeled by $\theta_{\sigma a}$. Note that the number of fresh states $w_{\sigma a}$ is countable (as the number of such transitions is countable) and therefore we have countably many success actions. Let Θ be the set of all such success actions and let $\mathcal{T}_{\mathsf{tm}}'$ denote

the modified version of $\mathcal{T}_{\mathsf{tm}}$. We assume that scheduler $\mathcal{D}'$ is defined on $\mathcal{T}'_{\mathsf{tm}}|\mathcal{P}'_1$ such that $\mathcal{D}'$ decides as $\mathcal{D} \in \mathsf{HR}(\mathcal{T}_{\mathsf{tm}}|\mathcal{P}'_1)$ does but chooses the success action θ_σ with probability one in w_σ (for all fresh states w_σ). Then for all $\sigma = a_1 a_2, \ldots, a_n \in \mathsf{Act}^*$, $\sigma' = a_1 z_{a_1} a_2 z_{a_2} \ldots a_n z_{a_n}$ it holds that

$$\begin{aligned}
& \mathsf{te}_{\mathcal{P}'_1}^{(\mathcal{T}'_{\mathsf{tm}},\mathcal{D}')}(\theta_\sigma, t) \\
= \;& \mathsf{Pr}^{\mathcal{D}'}(\{\pi \in \mathsf{path}(\mathcal{P}'_1|\mathcal{T}'_{\mathsf{tm}}) \mid \exists i : \\
& \qquad trace(\pi{\downarrow_i}) = \sigma'\theta_\sigma, time(\pi{\downarrow_i}) \leq t\}) \\
= \;& 0.5^n \cdot \mathsf{Pr}^{\mathcal{D}}(\{\pi \in \mathsf{path}(\mathcal{P}'_1|\mathcal{T}_{\mathsf{tm}}) \mid \exists i : \\
& \qquad trace(\pi{\downarrow_i}) = \sigma', time(\pi{\downarrow_i}) \leq t\}) \\
= \;& 0.5^n \cdot \mathsf{tr}^{\mathcal{D}}_{\mathcal{P}'_1|\mathcal{T}_{\mathsf{tm}}}(\sigma', t)
\end{aligned}$$

if $\theta_\sigma \in \Theta$. If $\sigma \in \mathsf{Act}^*$ is such that $\theta_\sigma \notin \Theta$ then $\mathsf{tr}^{\mathcal{D}}_{\mathcal{P}'_1|\mathcal{T}_{\mathsf{tm}}}(\sigma', t) = 0$.

Moreover, since $\mathcal{P}'_1 =_{\mathsf{te}} \mathcal{P}'_2$ there exists $\mathcal{E}' \in \mathsf{HR}(\mathcal{P}'_2|\mathcal{T}'_{\mathsf{tm}})$ such that for all $\theta_\sigma \in \Theta$, $t \geq 0$

$$\mathsf{te}_{\mathcal{P}'_1}^{(\mathcal{T}'_{\mathsf{tm}},\mathcal{D}')}(\theta_\sigma, t) \;=\; \mathsf{te}_{\mathcal{P}'_2}^{(\mathcal{T}'_{\mathsf{tm}},\mathcal{E}')}(\theta_\sigma, t).$$

Now, let $\mathcal{E}$ be a scheduler for $\mathcal{P}'_2|\mathcal{T}_{\mathsf{tm}}$ which decides as $\mathcal{E}'$ does (except that $\mathcal{E}$ is not defined on path fragments containing a state w_σ). But then for all $\sigma = a_1 a_2, \ldots, a_n \in \mathsf{Act}^*$, $\sigma' = a_1 z_{a_1} a_2 z_{a_2} \ldots a_n z_{a_n}$, $t \geq 0$

$$\begin{aligned}
\mathsf{tr}^{\mathcal{D}}_{\mathcal{P}'_1|\mathcal{T}_{\mathsf{tm}}}(\sigma', t) \;&=\; \tfrac{1}{0.5^n} \cdot \mathsf{te}_{\mathcal{P}'_1}^{(\mathcal{T}'_{\mathsf{tm}},\mathcal{D}')}(\theta_\sigma, t) \\
&=\; \tfrac{1}{0.5^n} \cdot \mathsf{te}_{\mathcal{P}'_2}^{(\mathcal{T}'_{\mathsf{tm}},\mathcal{E}')}(\theta_\sigma, t) \qquad\qquad \text{(A.15)} \\
&=\; \mathsf{tr}^{\mathcal{E}}_{\mathcal{P}'_2|\mathcal{T}_{\mathsf{tm}}}(\sigma', t).
\end{aligned}$$

if $\theta_\sigma \in \Theta$. If $\sigma \in \mathsf{Act}^*$ is such that $\theta_\sigma \notin \Theta$ we get

$$\mathsf{tr}^{\mathcal{D}}_{\mathcal{P}'_1|\mathcal{T}_{\mathsf{tm}}}(\sigma', t) = \mathsf{tr}^{\mathcal{E}}_{\mathcal{P}'_2|\mathcal{T}_{\mathsf{tm}}}(\sigma', t) = 0$$

which completes the final step.

The remainder of the proof is as follows: We define a tm-scheduler $\mathcal{F}_2$ for $\mathcal{P}_2$ in a similar way as $\mathcal{D}$ is defined for $\mathcal{P}'_1|\mathcal{T}_{\mathsf{tm}}$ (given $\mathcal{F}_1$) in the third step. More precisely, $\mathcal{F}_2$ resolves the remaining nondeterminism of $\mathcal{P}_2$ (while $\mathcal{P}_2$ is tested with timer settings tm) in the same way as $\mathcal{E}$ resolves nondeterminism in $\mathcal{P}'_2|\mathcal{T}_{\mathsf{tm}}$. As before, it follows by induction that $\mathsf{tr}^{\mathcal{E}}_{\mathcal{P}'_2|\mathcal{T}_{\mathsf{tm}}}$ and $\mathsf{de}^{(\mathsf{tm},\mathcal{F}_2)}_{\mathcal{P}_2}$ assign the same probability to trace observations. We derive for all $\sigma = a_1 a_2, \ldots, a_n \in \mathsf{Act}^*$, $\sigma' = a_1 z_{a_1} a_2 z_{a_2} \ldots a_n z_{a_n}$

$$\begin{aligned}
& \mathsf{tr}^{\mathcal{E}}_{\mathcal{P}'_2|\mathcal{T}_{\mathsf{tm}}}(\sigma', t) \\
= \; & \mathsf{Pr}^{\mathcal{E}}(\{\pi \in \mathsf{path}(\mathcal{P}'_2|\mathcal{T}_{\mathsf{tm}}) \mid \exists i : trace(\pi{\downarrow_i}) = \sigma', time(\pi{\downarrow_i}) \leq t\}) \\
= \; & \mathsf{Pr}^{(\mathsf{tm},\mathcal{F}_2)}(\{\pi \in \mathsf{path}(\mathcal{P}_2, \mathsf{tm}) \mid \exists i : trace(\pi{\downarrow_i}) = \sigma, time(\pi{\downarrow_i}) \leq t\} \\
= \; & \mathsf{de}^{(\mathsf{tm},\mathcal{F}_2)}_{\mathcal{P}_2}(\sigma, t).
\end{aligned} \tag{A.16}$$

Combining Equation A.14, A.15 and A.16 yields

$$\mathsf{de}^{(\mathsf{tm},\mathcal{F}_1)}_{\mathcal{P}_1}(\sigma, t) = \mathsf{tr}^{\mathcal{D}}_{\mathcal{P}'_1|\mathcal{T}_{\mathsf{tm}}}(\sigma', t) = \mathsf{tr}^{\mathcal{E}}_{\mathcal{P}'_2|\mathcal{T}_{\mathsf{tm}}}(\sigma', t) = \mathsf{de}^{(\mathsf{tm},\mathcal{F}_2)}_{\mathcal{P}_2}(\sigma, t), \tag{A.17}$$

for all $\sigma \in \mathsf{Act}^*, t \geq 0$, and the proof is complete. $\square$