A Definition Scheme for Quantitative Bisimulation

Diego Latella & Mieke Massink

Formal Methods and Tools
CNR/ISTI, Pisa, Italy

Erik de Vink*

Department of Mathematics and Computer Science, TU/e
Eindhoven, the Netherlands
Centrum voor Wiskunde en Informatica
Amsterdam, the Netherlands

Abstract *FuTS*, state-to-function transition systems are generalizations of labeled transition systems and of familiar notions of quantitative semantical models as continuous-time Markov chains, interactive Markov chains, and Markov automata. A general scheme for the definition of a notion of strong bisimulation associated with a *FuTS* is proposed. It is shown that this notion of bisimulation for a *FuTS* coincides with the coalgebraic notion of behavioral equivalence associated to the functor on **Set** given by the type of the *FuTS*. For a series of concrete quantitative semantical models the notion of bisimulation as reported in the literature is proven to coincide with the notion of quantitative bisimulation obtained from the scheme. The comparison includes models with orthogonal behaviour, like interactive Markov chains, and with multiple levels of behavior, like Markov automata. As a consequence of the general result relating *FuTS* bisimulation and behavioral equivalence we obtain, in a systematic way, a coalgebraic underpinning of all quantitative bisimulations discussed.

Keywords quantitative automata, state-to-function transition system, bisimulation

1 Introduction

State-to-Function Labeled Transition Systems (*FuTS*) have been introduced in [11] as a general framework for the formal definition of the semantics of process calculi, in particular of stochastic process calculi (*SPC*), as an alternative to the classical approach, based on labelled transition systems (*LTS*). In *LTS*, a transition is a triple (s, α, s') where s and α are the source state and the label of the transition, respectively, while s' is the target state reached from s via a transition labeled with α. In *FuTS*, a transition is a triple of the form (s, α, φ). The first and second component are the source state and the label of the transition, as in *LTS*, while the third component φ is a *continuation function* (or simply a *continuation* in the sequel), which associates a value from an appropriate semiring with each and every state s'. If φ maps s' to the 0 element of the semiring, then state s' cannot be reached from s via this transition. A non-zero value for a state s' represents a quantity associated with the jump of the system from s to s'. For instance, for continuous-time Markov chains (*CTMC*), the model underlying prominent *SPC*, this quantity is the rate of the negative exponential distribution characterizing the time for the execution of the action represented by α, necessary to reach s' from s via the transition.

We note that for the coalgebraic treatment itself of *FuTS* we propose here it is not necessary for the co-domain of continuations to be semirings; working with monoids would be sufficient. However, the richer structure of semirings is convenient, if not essential, when using continuations and their operators in the formal definition of the *FuTS* semantics of *SPC*. The use of continuations provides a clean and simple solution to the transition multiplicity problem and makes *FuTS* with $\mathbb{R}_{\geqslant 0}$ as semiring particularly suited for *SPC* semantics. We refer to [11] for a thorough discussion on the use of *FuTS* as the underlying semantics model for *SPC*—including those where (continuous-time) stochastic behaviour is integrated

---

*Corresponding author, email evink@win.tue.nl.

with non-determinism as well as with discrete probability distributions over behaviours—and for their comparison with other recent approaches to a uniform treatment of *SPC* semantics, e.g. Rated Transition Systems [19], Rate Transition Systems [10], Weighted Transition Systems [18], and *ULTraS* [6].

In [24] we proposed a coalgebraic view of *FuTS* involving functors $\mathcal{FS}(\cdot, \mathcal{R})$, for $\mathcal{R}$ a semiring, in place to deal with quantities. Here, for a set of states S, we use $\mathcal{FS}(S, \mathcal{R})$ to denote the set of all finitely supported functions from S to $\mathcal{R}$. Discrete probability distributions are a typical example of such a function space. Note, the functor $\mathcal{FS}(\cdot, \mathcal{R})$ over a monoid or semiring $\mathcal{R}$, unifies the finite subsets functor $\mathcal{P}_\omega(\cdot)$ and the discrete probability distributions functor *Distr*$(\cdot)$, choosing the semirings $\mathbb{B}$ and $\mathbb{R}_{\geqslant 0}$, respectively (and requiring restrictions on the continuations for the latter). Coalgebras, also built from the finite support functor, are of interest because they come equipped, under mild conditions, with a canonical notion of 'bisimulation' known as behavioral equivalence [28]. Our earlier work presented a notion of bisimulation for *FuTS* and we proved a correspondence result stating that *FuTS* bisimilarity coincides with the behavioral equivalence of the associated functor. We applied the framework to prominent *SPC* like *PEPA* [17] and a language for Interactive Markov Chains (*IMC*) of [15], thus providing coalgebraic justification of the equivalences of these calculi. In [26] an approach similar to ours has been applied to the *ULTraS* model, a model which shares some features with simple *FuTS*. In *ULTraS* posets are used instead of semirings, although a monoidal structure is then implicitly assumed when process equivalences are taken into consideration [6].

An interesting direction of research combining coalgebra and quantities investigates various types of weighted automata, including linear weighted automata, and associated notions of bisimulation and languages, as well as algorithms for these notions [7, 18]. Klin considers weighted transition systems, labelled transition systems that assign a weight to each transition and develops Weighted GSOS, a meta-syntactic framework for defining well-behaved weighted transition systems. For commutative monoids the notion of a weighted transition system compares with our notion of a *FuTS*, and, when cast in the coalgebraic setting, the associated concept of bisimulation coincides with behavioral equivalence. Weights of transitions of weighted transition systems are computed by induction on the syntax of process terms and by taking into account the contribution of all those GSOS rules that are triggered by the relevant (apparent) weights. Note that such a set of rules is finite. So, in a sense, the computation of the weights is distributed among (the instantiations of) the relevant rules with intermediate results collected and integrated in the final weight. In [26] a general GSOS specification format is presented which allows for a 'syntactic' treatment of continuations involving so-called *weight functions*. A comparison of a wide range of probabilistic transition systems focusing on coalgebraic bisimulation is reported in [5, 30], which provide hierarchy relating types of transition systems via natural embeddings. In [22] the investigation on the relationship between *FuTS* bisimilarity and behavioural equivalence, and also coalgebraic bisimilarity is presented. In particular, it is shown that the functor type involved preserves weak pullbacks when the underlying semiring satisfies the zero-sum property.

So far, all the unifying approaches to modeling of *SPC* discussed above restrict to a single layer of quantities. Although for *IMC* orthogonal transition relations, hence products of continuation sets need to be considered, overall no nesting of the construction with finitely supported functions is allowed. The problem appears to lie, at least for *FuTS*, in identifying an appropriate notion of bisimulation. Bisimulation for probabilistic systems is traditionally based on equivalence classes [21, 3, 9] with a lifting operator from states to probability distributions as a main ingredient. In the present paper, we extend the results reported in [24] by taking this ingredient of lifting into account systematically, thus also catering for repeated lifting. In line with the coalgebraic paradigm, the type of the *FuTS*, hence not its transitions, decides the way equivalence classes need to be lifted. The generalization paves the way to deal with more intricate interactions of qualitative and quantitative behaviour. For instance, now we are

capable to deal uniformly with Probabilistic Automata (*PA*), see e.g. [29, 14], and Markov Automata (*MA*), cf. [13, 12, 33], which do not fit in the unifying treatments mentioned above. Thus, the present paper constitutes an additional step forward to implementing the aim of a uniform treatment of semantic models for quantitative process calculi and associated notion of strong bisimulation, but avoiding implicit transition multiplicity and/or explicit transition decoration (see [11]). In addition, we have coalgebra interpretations and bisimulation correspondence results at our disposal as a yardstick for justifying particular process equivalence as being the natural ones.

The paper is structured as follows: Section 2 introduces notation and concepts related to the nesting of constructs of the form $\mathcal{FS}(\cdot, \mathcal{R})$, for a semiring $\mathcal{R}$, and briefly discusses coalgebraic notions. Section 3 recalls the notion of a *FuTS*, distinguishing simple, combined, nested and general *FuTS*. The notion of bisimulation for a *FuTS*, the scheme of defining quantitative bisimulation and the comparison of the latter with behavioral equivalence is addressed too. Section 4 presents the treatment of *LTS*, of *CTMC* and of *IMC* with *FuTS*. In particular, the concrete notions of strong bisimulation for these semantical models are related to *FuTS* bisimulation. Section 5 continues this, now for *PA* and *MA*, semantical models that involve nesting. Also for these models concrete bisimulation and *FuTS* bisimulation, as obtained from the general scheme, are shown to coincide. Section 6 wraps up with concluding remarks.

2   Preliminaries

A semiring $\mathcal{R}$ is tuple $(R, +, 0, \cdot, 1)$ with $(R, +, 0)$ a commutative monoid with zero element 0, $(R, \cdot, 1)$ a monoid with identity element 1 such that the multiplication $\cdot$ distributes over the addition $+$, and the zero element 0 annihilates R, i.e. $0 \cdot r = 0 = r \cdot 0$. A function $f : X \to R$ from a set X to the carrier of a semiring $\mathcal{R}$ is said to be of finite support if the set $\{ x \in X \mid f(x) \neq 0 \}$ is finite. We use $\mathcal{FS}(X, \mathcal{R})$ to denote the set of all functions from X to $\mathcal{R}$ of finite support. The shorthand $\mathcal{FS}(X, \mathcal{R}_1, \ldots, \mathcal{R}_n)$, for a set X and semirings $\mathcal{R}_1, \ldots, \mathcal{R}_n$, for $n \geqslant 0$, is given by $\mathcal{FS}(X) = X$, and $\mathcal{FS}(X, \mathcal{R}_1, \ldots, \mathcal{R}_n, \mathcal{R}_{n+1}) = \mathcal{FS}(\mathcal{FS}(X, \mathcal{R}_1, \ldots, \mathcal{R}_n), \mathcal{R}_{n+1})$. We let $\mathcal{FS}(X, \mathcal{R})^{\mathcal{L}}$ denote the set of functions from $\mathcal{L}$ to $\mathcal{FS}(X, \mathcal{R})$ and we extend the notation to $\mathcal{FS}(X, \mathcal{R}_1, \ldots, \mathcal{R}_n)^{\mathcal{L}}$ in the obvious way.

For a set X, a semiring $\mathcal{R}$, a set of finitely supported functions $F \subseteq \mathcal{FS}(X, \mathcal{R})$, and an equivalence relation E on X, we define the relation $\mathcal{LT}(E, \mathcal{R})_F$ with respect to F by $\mathcal{LT}(E, \mathcal{R})_F(\varphi, \psi) \iff \varphi[B] = \psi[B]$ for all $B \in X/E$, for $\varphi, \psi \in F$. Here $\chi[A]$ denotes $\sum_{a \in A} \chi(a)$, for $\chi \in \mathcal{FS}(X, \mathcal{R})$ and $A \subseteq X$. Note that $\mathcal{LT}(E, \mathcal{R})_F$ is an equivalence relation on F. We use $\mathcal{LT}(E, \mathcal{R})$ to denote the lifting of E with respect to $\mathcal{FS}(X, \mathcal{R})$ itself. For a sequence of semirings $\mathcal{R}_1, \ldots, \mathcal{R}_n$, we define the relation $\mathcal{LT}(E, \mathcal{R}_1, \ldots, \mathcal{R}_n)$ on $\mathcal{FS}(X, \mathcal{R}_1, \ldots, \mathcal{R}_n)$ by $\mathcal{LT}(E) = E$, and $\mathcal{LT}(E, \mathcal{R}_1, \ldots, \mathcal{R}_{n+1})(\varphi, \psi) \iff \varphi[C] = \psi[C]$ for all $C \in \mathcal{FS}(X, \mathcal{R}_1, \ldots, \mathcal{R}_n)/\mathcal{LT}(E, \mathcal{R}_1, \ldots, \mathcal{R}_n)$ and $\varphi, \psi \in \mathcal{FS}(X, \mathcal{R}_1, \ldots, \mathcal{R}_{n+1})$. By induction on n one establishes that $\mathcal{LT}(E, \mathcal{R}_1, \ldots, \mathcal{R}_n)$ is an equivalence relation too.

If E and F are binary relations of the sets X and Y, respectively, we define the relation $E \times F \subseteq (X \times Y) \times (X \times Y)$ by $E \times F = \{ (\langle x_1, y_1 \rangle, \langle x_2, y_2 \rangle) \mid E(x_1, x_2) \wedge F(y_1, y_2) \}$. It holds that $E \times F$ is an equivalence relation if E and F are.

For a commutative monoid, hence for a semiring or a field, the functor $\mathcal{FS}(\cdot, \mathcal{R}) : \mathbf{Set} \to \mathbf{Set}$, on the category $\mathbf{Set}$ of sets and functions, assigns to a set X the set of all finitely supported functions $\mathcal{FS}(X, \mathcal{R})$, and to a function $f : X \to Y$ the function $\mathcal{FS}(f, \mathcal{R}) : \mathcal{FS}(X, \mathcal{R}) \to \mathcal{FS}(Y, \mathcal{R})$ given by

$$\mathcal{FS}(f, \mathcal{R})(\varphi)(y) = \sum_{x \in f^{-1}(y)} \varphi(x)$$

Note, since φ is assumed to have a finite support, the summation at the right side in $\mathcal{R}$ may be infinite, but is still well-defined because of commutativity of $+$ on $\mathcal{R}$. Also note, only addition of $\mathcal{R}$ is used above.

However, in concrete situations, in particular when modeling the parallel operator for *SPC*, multiplication of $\mathcal{R}$ is needed as well, cf. [24, 22].

A pair of a set and a mapping (X, α) is called a coalgebra of a functor F on **Set** if $\alpha : X \to FX$. A mapping $f : X \to Y$ is a coalgebra homomorphism for two coalgebras (X, α) and (Y, β), if $Ff \circ \alpha = \beta \circ f$. A final coalgebra of F is a coalgebra (Ω, ω) of F such that for every coalgebra X there exists a unique coalgebra homomorphism $[\![\cdot]\!]_F^X : X \to \Omega$. A **Set** functor F is accessible, if it preserves κ-filtered colimits, for some regular cardinal κ. Typically, one uses the following characterization of accessibility: every element of FX, for any set X, lies in the image of some subset $Y \subseteq X$ of less than κ elements [1]. In essence, familiar functors like the finite powerset functor $\mathcal{P}_\omega(\cdot)$ and the discrete probability distribution functor $Distr(\cdot)$ are accessible. Also, their generalization used in this paper, viz. the finitely supported functions functor $\mathcal{FS}(\cdot, \mathcal{R})$ [18, 7] on the category **Set** is accessible as well. Accessibility is preserved by exponentiation and products. Therefore, all functors appearing in the sequel are accessible.

The appealing feature of an accessible functor on **Set** is that it possesses a final coalgebra. For an accessible **Set** functor F it follows that the mapping $[\![\cdot]\!]_F^X : X \to \Omega$ is well-defined, for each F-coalgebra (X, α). Now, two elements $x, y \in X$ are called *behaviorally equivalent* for the functor F, notation $x \approx_F y$, if $[\![x]\!]_F^X = [\![y]\!]_F^X$. In a way, if $x \approx_F y$ then x and y can be identified according to F. Although $[\![\cdot]\!]_F^X$ is an F-coalgebra homomorphism, i.e. $\omega \circ [\![\cdot]\!]_F^X = F([\![\cdot]\!]_F^X) \circ \alpha$, one can argue that in fact the functor F determines which elements of X are behavioral equivalent.

3   State-to-function transition systems

We start off with a formal definition of a state-to-function transition system (*FuTS*), we introduce the scheme for defining the notion of bisimulation of a *FuTS*, and we relate *FuTS* bisimulation to behavioral equivalence for the type functor of a *FuTS*.

Definition 1. *A FuTS for a sequence of label sets* $\mathcal{L}_1, \ldots, \mathcal{L}_n$ *and a sequence of sequences of semirings* $(\mathcal{R}_{1,j})_{j=1}^{m_1}, \ldots, (\mathcal{R}_{n,j})_{j=1}^{m_n}$*, for* $n, m_1, \ldots, m_n > 0$*, is a pair* $\mathcal{X} = (X, \theta)$ *of a set* X *and a mapping*

$$\theta : X \to \mathcal{FS}(X, \mathcal{R}_{1,1}, \ldots, \mathcal{R}_{1,m_1})^{\mathcal{L}_1} \times \cdots \times \mathcal{FS}(X, \mathcal{R}_{n,1}, \ldots, \mathcal{R}_{n,m_n})^{\mathcal{L}_n}$$

In the sequel, when $n > 1$, we usually represent the mapping θ as a tuple of mappings $\langle \theta_1, \ldots, \theta_n \rangle$ with $\theta_i : X \to \mathcal{FS}(X, \mathcal{R}_{i,1}, \ldots, \mathcal{R}_{i,m_i})^{\mathcal{L}_i}$. In the present paper we focus on the class of *deterministic FuTS*, namely models where the transition relation is a function, as in Definition 1. Note, this excludes by no means the treatment of non-deterministic systems. Below and in [11, 24] it has been shown that the class of *FuTS* given by the definition above is sufficiently rich to deal with all the major stochastic process description languages and their underlying semantic models.

In the next two sections we will discuss several examples of *FuTS* coming in various flavors. In particular we distinguish the cases of a simple *FuTS*, of a combined *FuTS*, and of a nested *FuTS*. A *FuTS* of the form $\mathcal{X} = (X, \theta)$ with $\theta : X \to \mathcal{FS}(X, \mathcal{R})^{\mathcal{L}}$, thus $n = 1$, $m_1 = 1$, is called a *simple FuTS*. We will see that the familiar labeled transition systems (*LTS*) over an action set $\mathcal{A}$ are simple *FuTS* for label set $\mathcal{A}$ and semiring $\mathbb{B}$, i.e. an *LTS* with set of states S over $\mathcal{A}$ corresponds to a simple *FuTS* with transition function $\theta : S \to \mathcal{FS}(S, \mathbb{B})^{\mathcal{A}}$. Likewise, discrete-time and continuous-time Markov chains (*DTMC, CTMC*) are simple *FuTS* over a degenerate label set Δ and semiring $\mathbb{R}_{\geqslant 0}$. Putting $\Delta = \{\delta\}$, a Markov chain with set of states S can be identified with a simple *FuTS* with transition function $\theta : S \to \mathcal{FS}(S, \mathbb{R}_{\geqslant 0})^{\Delta}$. For a *DTMC* we will have $\theta(s)(\delta)[S] = 1$.

A *FuTS* of the form $\mathfrak{X} = (X, \theta)$ with $\theta = \langle \theta_1, \dots, \theta_n \rangle$ and $\theta_i : X \to \mathcal{FS}(X, \mathcal{R}_i)^{\mathcal{L}_i}$, for $i = 1 \dots n$, thus $m_1, \dots, m_n = 1$, is called a *combined FuTS*. Interactive Markov chains (*IMC*, cf. [15]) are prominent examples of combined *FuTS*. An *IMC* with states from S and action set $\mathcal{A}$ can be seen as a combined *FuTS* for the label sets $\mathcal{A}$ and Δ, and semirings $\mathbb{B}$ and $\mathbb{R}_{\geqslant 0}$ with a pair of transition functions $\langle \theta_1, \theta_2 \rangle$. Here $\theta_1 : S \to \mathcal{FS}(S, \mathbb{B})^{\mathcal{A}}$ captures the interactive component of the *IMC* and $\theta_2 : S \to \mathcal{FS}(S, \mathbb{R}_{\geqslant 0})^{\Delta}$ captures the Markovian component.

Finally, a *FuTS* of the form $\mathfrak{X} = (X, \theta)$ with $\theta : X \to \mathcal{FS}(X, \mathcal{R}_1, \dots, \mathcal{R}_m)^{\mathcal{L}}$ is called a *nested FuTS*. Below we will argue that probabilistic automata (*PA*, cf. [29, 14]) over an action set $\mathcal{A}$ are nested *FuTS* for the label set $\mathcal{A}$ and semirings $\mathbb{R}_{\geqslant 0}$ and $\mathbb{B}$. A *PA* with states from S and actions from $\mathcal{A}$ induces a two-level nested *FuTS* with transition function $\theta : S \to \mathcal{FS}(S, \mathbb{R}_{\geqslant 0}, \mathbb{B})^{\mathcal{A}}$, or more explicitly $\theta : S \to \mathcal{FS}(\mathcal{FS}(S, \mathbb{R}_{\geqslant 0}), \mathbb{B})^{\mathcal{A}}$. For continuations $\pi \in \mathcal{FS}(S, \mathbb{R}_{\geqslant 0})$ involved, it will hold that $\pi[S] = 1$.

Most automata in the setting of quantitative process languages fall in the three special types of *FuTS* mentioned (simple, combined or nested). An important semantic model not captured is that of Markov automaton (MA, cf. [13, 12, 33]). A Markov automaton can be seen as a 'general' *FuTS*, i.e. a *FuTS* that is not of one of the distinguished types. More precisely, an *MA* with set of states S and action set $\mathcal{A}$ can be represented as a *FuTS* for the label sets $\mathcal{A}$ and Δ and sequences of the two semirings $\mathbb{R}_{\geqslant 0}, \mathbb{B}$ and of only one semiring $\mathbb{R}_{\geqslant 0}$, thus having a pair of transition functions $\langle \theta_1, \theta_2 \rangle : S \to \mathcal{FS}(\mathcal{FS}(S, \mathbb{R}_{\geqslant 0}), \mathbb{B})^{\mathcal{A}} \times \mathcal{FS}(S, \mathbb{R}_{\geqslant 0})^{\Delta}$. Here, θ_1 represents the so-called immediate transition relation, while θ_2 represents the so-called timed transition relation.

Next we define a notion of (strong) bisimulation $\simeq_{\mathfrak{X}}$ for a *FuTS* $\mathfrak{X} = (X, \theta)$. By the definition below, a bisimulation relation E is an equivalence relation on the set of states X. The relation E on X is then lifted to an equivalence relation $\mathcal{T}(E)$ on $\mathcal{T}(X)$, invoking the so-called type $\mathcal{T}$ of the *FuTS* (formally given in Definition 3). For E to be a *FuTS* bisimulation we require, that $E(x, y)$ for states x and y implies $\mathcal{T}(E)(\theta(x), \theta(y))$.

Definition 2. *Let $\mathfrak{X} = (X, \theta)$ be a FuTS with transition function*

$$\theta : X \to \mathcal{FS}(X, \mathcal{R}_{1,1}, \dots, \mathcal{R}_{1,m_1})^{\mathcal{L}_1} \times \cdots \times \mathcal{FS}(X, \mathcal{R}_{n,1}, \dots, \mathcal{R}_{n,m_n})^{\mathcal{L}_n}$$

An equivalence relation $E \subseteq X \times X$ is called a bisimulation for $\mathfrak{X}$ if

$$E(x, y) \implies \mathcal{LT}(E, \mathcal{R}_{1,1}, \dots, \mathcal{R}_{1,m_1})^{\mathcal{L}_1} \times \cdots \times \mathcal{LT}(E, \mathcal{R}_{n,1}, \dots, \mathcal{R}_{n,m_n})^{\mathcal{L}_n}(\theta(x), \theta(y))$$

for all $x, y \in X$. Two states $x, y \in X$ are $\mathfrak{X}$-bisimilar, written $x \simeq_{\mathfrak{X}} y$, if $E(x, y)$ for a bisimulation E for $\mathfrak{X}$.

Recall, since E is assumed to be an equivalence relation on X, we have that $\mathcal{LT}(E, \mathcal{R}_{i,1}, \dots, \mathcal{R}_{i,m_i})^{\mathcal{L}_i}$ is an equivalence relation on $\mathcal{FS}(X, \mathcal{R}_{i,1}, \dots, \mathcal{R}_{i,m_i})^{\mathcal{L}_i}$, for all i. Moreover, by Definition 1 we have $m_i > 0$ for all i. Expanding the definition of the outer $\mathcal{LT}(\cdot, \mathcal{R}_{i,m_i})$ for the relation involved, yields that if two states are equivalent, i.e. $E(x, y)$, then evaluating $\theta_i(x)(\ell)[C]$ and $\theta_i(y)(\ell)[C]$ amounts to the same, for all labels $\ell \in \mathcal{L}_i$ and equivalence classes C of $\mathcal{LT}(E, \mathcal{R}_{i,1}, \dots, \mathcal{R}_{i,m_i-1})$, for all i, i.e. $\theta_i(x)(\ell)[C] = \theta_i(y)(\ell)[C]$, for $i = 1, \dots, n$.

For simple or combined *FuTS* the scheme is all straightforward since $\mathcal{LT}(E, \mathcal{R}_{i,1}, \dots, \mathcal{R}_{i,m_i-1})$ is just E, cf. [24]. However, when the codomain of the *FuTS* involves nested applications of the $\mathcal{FS}(\cdot, \mathcal{R})$ operator, $\mathcal{FS}(X, \mathcal{R}_{i,1}, \dots, \mathcal{R}_{i,m_i-1})$ is of a higher functional level than the set X itself. Therefore, we need to push the relation E up, so to speak pushing it through the component operators of the codomain.

In the sequel we will see several examples of quantitative automata and their notion of bisimulation from the literature to coincide with their *FuTS* representation and *FuTS* bisimulation of Definition 2. However, the point is that *FuTS* bisimulation can also be captured coalgebraically. In fact, *FuTS* bisimulation and so-called behavioral equivalence [20, 32] are the same. Therefore, by relating a quantitative execution model like a *CTMC*, *IMC*, or *MA* with a suitable *FuTS*, immediately provides coalgebraic justification of the specific notion of bisimulation as the natural (strong) process equivalence.

Definition 3. *A FuTS* $\mathcal{X} = (X, \theta)$ *with transition function*

$$\theta : X \to \mathcal{FS}(X, \mathcal{R}_{1,1}, \ldots, \mathcal{R}_{1,m_1})^{\mathcal{L}_1} \times \cdots \times \mathcal{FS}(X, \mathcal{R}_{n,1}, \ldots, \mathcal{R}_{n,m_n})^{\mathcal{L}_n}$$

is called a FuTS of type $\mathcal{T}$*, for the **Set**-functor* $\mathcal{T}$ *given by*

$$\mathcal{T} = \mathcal{FS}(\ldots \mathcal{FS}(\cdot, \mathcal{R}_{1,1}) \ldots, \mathcal{R}_{1,m_1})^{\mathcal{L}_1} \times \cdots \times \mathcal{FS}(\ldots \mathcal{FS}(\cdot, \mathcal{R}_{n,1}) \ldots, \mathcal{R}_{n,m_n})^{\mathcal{L}_n}$$

Thus, if a *FuTS* $\mathcal{X}$ is of type $\mathcal{T}$ for a functor $\mathcal{T}$, then, in turn, $\mathcal{X}$ is a coalgebra of $\mathcal{T}$. Note, $\mathcal{T}$ is a composition of **Set**-functors: 'finite support' functors $\mathcal{FS}(\cdot, \mathcal{R})$, exponentiation functors $(\cdot)^{\mathcal{L}}$, and product functors $(\cdot) \times (\cdot)$. This restricted form gives rise to the following result.

Theorem 4. *If a functor* $\mathcal{T}$ *on **Set** is the type of a FuTS, then* $\mathcal{T}$ *possesses a final coalgebra.*

Proof. Functors of the form $\mathcal{FS}(\cdot, \mathcal{R})$ can be shown to be accessible using a standard argument, cf. [18, 7]. Accessibility is preserved by products, exponentiation and composition. It follows that $\mathcal{T}$ itself is accessible, and hence has a final coalgebra, see [1]. $\square$

Let $\mathcal{X} = (X, \theta)$ be a *FuTS* of type functor $\mathcal{T}$ and let $\Omega = (\Omega, \omega)$ denote the final coalgebra of $\mathcal{T}$. By finality of Ω there exists a unique $\mathcal{T}$-homomorphism $[\![\cdot]\!]_{\mathcal{X}}^{\mathcal{T}} : X \to \Omega$. Behavioral equivalence $\approx_{\mathcal{T}}$ is then defined as $x \approx_{\mathcal{T}} y$ iff $[\![x]\!]_{\mathcal{X}}^{\mathcal{T}} = [\![y]\!]_{\mathcal{X}}^{\mathcal{T}}$. We have the following result relating FuTS bisimilarity $\simeq_{\mathcal{X}}$ to behavioral equivalence $\approx_{\mathcal{T}}$ of the type functor $\mathcal{T}$.

Theorem 5 (correspondence theorem). *Let* $\mathcal{X} = (X, \theta)$ *be a FuTS of type* $\mathcal{T}$ *for the functor* $\mathcal{T}$ *on **Set**. Then it holds that* $x \simeq_{\mathcal{X}} y$ *iff* $x \approx_{\mathcal{T}} y$*, i.e. FuTS bisimulation and behavioral equivalence coincide.* $\square$

A restricted version of Theorem 5 was given in [24, 22]. The present theorem generalizes the result to also deal with nesting, a situation needed for the more advanced quantitative automata discussed in the sequel. The proof of the theorem is built on two lemmas. To smooth the presentation, we consider the lemmas only for non-product functors (i.e. choosing $n = 1$ in Definition 3). The extension to product functors is conceptually straightforward.

Recall, for $f : X \to Y$, the functor application $\mathcal{FS}(\ldots \mathcal{FS}(f, \mathcal{R}_1) \ldots, \mathcal{R}_n)^{\mathcal{L}}$ to f is a function from the set $\mathcal{FS}(X, \mathcal{R}_1, \ldots, \mathcal{R}_n)^{\mathcal{L}}$ to the set $\mathcal{FS}(Y, \mathcal{R}_1, \ldots, \mathcal{R}_n)^{\mathcal{L}}$ with $\mathcal{FS}(f, \mathcal{R}_1)(\varphi)(y) = \sum_{x \in f^{-1}(y)} \varphi(x)$ for $\varphi \in \mathcal{FS}(X, \mathcal{R}_1)$ and $y \in Y$, and

$$\mathcal{FS}(\ldots \mathcal{FS}(f, \mathcal{R}_1) \ldots, \mathcal{R}_n)(\Phi)(\ell)(\psi) = \sum_{\varphi \in \mathcal{FS}(\ldots \mathcal{FS}(f, \mathcal{R}_1) \ldots, \mathcal{R}_{n-1})^{-1}(\psi)} \Phi(\ell)(\phi)$$

for $\Phi \in \mathcal{FS}(X, \mathcal{R}_1, \ldots, \mathcal{R}_n)^{\mathcal{L}}$, $\ell \in \mathcal{L}$, $\psi \in \mathcal{FS}(Y, \mathcal{R}_1, \ldots, \mathcal{R}_{n-1})$, and resulting value in $\mathcal{R}_n$.

Lemma 1. *Let $\mathfrak{X} = (X, \theta)$ be a FuTS of type $\mathfrak{T} = \mathcal{FS}(\ldots \mathcal{FS}(\cdot, \mathcal{R}_1) \ldots, \mathcal{R}_n)^{\mathcal{L}}$. If E is an equivalence relation on X, then there exists a mapping $\theta_E : X/E \to \mathcal{FS}(X/E, \mathcal{R}_1, \ldots, \mathcal{R}_n)^{\mathcal{L}}$ such that $\mathfrak{X}_E = (X/E, \theta_E)$ is a coalgebra of the functor $\mathfrak{T}$, and the canonical mapping $\varepsilon : X \to X/E$ is a $\mathfrak{T}$-homomorphism.*

Sketch of proof. Define a coalgebra structure θ_E on X/E by putting

$$\theta_E([x]_E)(\ell)(\bar{z}_{n-1}) = \sum\nolimits_{z_{n-1} \in \mathcal{FS}(\varepsilon, \mathcal{R}_1, \ldots, \mathcal{R}_{n-1})^{-1}(\bar{z}_{n-1})} \theta(x)(\ell)(z_{n-1})$$

for $x \in X$, $\ell \in \mathcal{L}$, $\bar{z}_{n-1} \in \mathcal{FS}(X/E, \mathcal{R}_1, \ldots, \mathcal{R}_{n-1})$. The property

$$\mathcal{FS}(\varepsilon, \mathcal{R}_1, \ldots, \mathcal{R}_i)(z_i) = \mathcal{FS}(\varepsilon, \mathcal{R}_1, \ldots, \mathcal{R}_i)(z'_i) \iff \mathcal{LT}(E, \mathcal{R}_1, \ldots \mathcal{R}_i)(z_i, z'_i)$$

for $z_i, z'_i \in \mathcal{FS}(X, \mathcal{R}_1, \ldots, \mathcal{R}_i)$, $i = 1, \ldots, n$, can be proved by induction on i. Then, the property for n is the key ingredient to verify that $\mathcal{FS}(\varepsilon, \mathcal{R}_1, \ldots, \mathcal{R}_n)^{\mathcal{L}} \circ \theta = \theta_E \circ \varepsilon$, i.e. ε is a $\mathfrak{T}$-homomorphism. $\qquad\square$

From the lemma it follows that the final mapping $[\![\cdot]\!]_{\mathfrak{T}}^{\mathfrak{X}} : X \to \Omega$ factorizes through ε. Hence, if $\varepsilon(x) = \varepsilon(y)$, then $x \approx_{\mathfrak{T}} y$, proving half of Theorem 5. The reverse can be shown using the following result.

Lemma 2. *Let $\mathfrak{X} = (X, \theta)$ be a FuTS of type $\mathfrak{T} = \mathcal{FS}(\ldots \mathcal{FS}(\cdot, \mathcal{R}_1) \ldots, \mathcal{R}_n)^{\mathcal{L}}$. The relation $\approx_{\mathfrak{T}}$ on X is a bisimulation for the FuTS $\mathfrak{X}$.*

Sketch of proof. One first shows, by induction on i,

$$y_i \in [x_i]_{\mathcal{LT}(\approx_{\mathfrak{T}}, \mathcal{R}_1, \ldots, \mathcal{R}_i)} \iff \mathcal{FS}([\![\cdot]\!]_{\mathfrak{T}}^{\mathfrak{X}}, \mathcal{R}_1, \ldots, \mathcal{R}_i)(x_i) = \mathcal{FS}([\![\cdot]\!]_{\mathfrak{T}}^{\mathfrak{X}}, \mathcal{R}_1, \ldots, \mathcal{R}_i)(y_i)$$

for $x_i, y_i \in \mathcal{FS}(X, \mathcal{R}_1, \ldots, \mathcal{R}_i)$, and $i = 1, \ldots, n$. Using the above property for n, one next verifies

$$\omega([\![x]\!]_{\mathfrak{T}}^{\mathfrak{X}})(\ell)(w_{n-1}) = \sum\nolimits_{z_{n-1} \in \mathcal{FS}([\![\cdot]\!]_{\mathfrak{T}}^{\mathfrak{X}}, \mathcal{R}_1, \ldots, \mathcal{R}_{n-1})^{-1}(w_{n-1})} \theta(x)(\ell)(z_{n-1})$$

for $x \in X$, $\ell \in \mathcal{L}$, and $w_{n-1} \in \mathcal{FS}(\Omega, \mathcal{R}_1, \ldots, \mathcal{R}_{n-1})$. From this the 'transfer condition' of Definition 2 for $\approx_{\mathfrak{T}}$ follows. $\qquad\square$

By the lemma, if $x \approx_{\mathfrak{T}} y$, then there exists a bisimulation for the *FuTS* $\mathfrak{X}$, viz. the bisimulation $\approx_{\mathfrak{T}}$, that relates x and y. This proves the other direction of the correspondence theorem.

In the next two sections, we proceed to incorporate the major operational models used for the semantics of quantitative process languages in the *FuTS* framework. Theorem 5 confirms that *FuTS* bisimulation is a proper notion of process equivalence. Thus, as a consequence, a notion of process equivalence that coincides with *FuTS* bisimulation also coincides with its coalgebraic counterpart.

4   Quantitative transition systems as simple and combined *FuTS*

In this section we interpret standard labeled transition systems, continuous-time Markov chains and Hermann's interactive Markov chains as *FuTS* and show that their usual notion of bisimulation coincides with the notion of bisimulation of their associated *FuTS*.

4.1   Labeled transition systems

It is straightforward to see that an *LTS*, over a set of actions $\mathcal{A}$ and with a set of states S, can be modeled as a function $S \times \mathcal{A} \to (S \to \mathbb{B})$. However, arbitrary *LTS* do not fit in our set-up with finitely supported functions. So, our modeling of *LTS* here, similar as reported elsewhere, e.g. [19, 7], restricts to image-finite *LTS*.

Definition 6. *Fix a set $\mathcal{A}$ of actions.*

(a) *An image-finite LTS over $\mathcal{A}$ is a pair $\mathcal{L} = (S, \to_{\mathcal{L}})$ where S is a set of states, and $\to_{\mathcal{L}} \subseteq S \times \mathcal{A} \times S$ is the transition relation such that, for all $s \in S$, $a \in \mathcal{A}$, the set $\{\, s' \mid s \xrightarrow{a}_{\mathcal{L}} s' \,\}$ is finite.*

(b) *An equivalence relation $R \subseteq S \times S$ is called a bisimulation equivalence for the LTS $\mathcal{L} = (S, \to_{\mathcal{L}})$ if, for all $s, s', t \in S$, $a \in \mathcal{A}$ such that $R(s,t)$ and $s \xrightarrow{a}_{\mathcal{L}} s'$, there exists $t' \in S$ such that $t \xrightarrow{a}_{\mathcal{L}} t'$ and $R(s',t')$.*

(c) *Two states $s, t \in S$ in an LTS $\mathcal{L} = (S, \to_{\mathcal{L}})$ are called strongly bisimilar for LTS $\mathcal{L}$, if there exists a bisimulation equivalence R for $\mathcal{L}$ such that $R(s,t)$.*

An image-finite *LTS* $\mathcal{L} = (S, \to_{\mathcal{L}})$ over $\mathcal{A}$ induces a simple *FuTS* $\mathcal{F}(\mathcal{L}) = (S, \theta_{\mathcal{L}})$ over $\mathcal{A}$ and the semiring $\mathbb{B}$, if we define $\theta_{\mathcal{L}} : S \to \mathcal{FS}(S, \mathbb{B})^{\mathcal{A}}$ by $\theta_{\mathcal{L}}(s)(a)(s') \iff s \xrightarrow{a}_{\mathcal{L}} s'$, for all $s, s' \in S$, $a \in \mathcal{A}$. The next theorem will not come as an surprise. See [19, 7] for example, for a proof that the respective notions of bisimulation coincide for this specific case. In order to illustrate the general pattern of such a proof for *FuTS* we provide a proof here as well.

Theorem 7. *Let $\mathcal{L} = (S, \to_{\mathcal{L}})$ be an LTS. Then it holds that R is a bisimulation equivalence iff R is a FuTS bisimulation for $\mathcal{F}(\mathcal{L})$.*

Proof. The result follows almost directly from the definitions. We have, for $s, t \in S$,

$$
\begin{aligned}
&\mathcal{LT}(R, \mathbb{B})^{\mathcal{A}}(s,t) \\
&\iff \quad \forall a \in \mathcal{A} \, \forall C \in S/R \colon \textstyle\sum_{u \in C} \theta_{\mathcal{L}}(s)(a)(u) = \sum_{u \in C} \theta_{\mathcal{L}}(t)(a)(u) \quad (\text{definition } \mathcal{LT}(R, \mathbb{B})^{\mathcal{A}}) \\
&\iff \quad \forall a \in \mathcal{A} \, \forall C \in S/R \colon \exists u \in C \colon s \xrightarrow{a}_{\mathcal{L}} u \Leftrightarrow \exists u \in C \colon t \xrightarrow{a}_{\mathcal{L}} u \quad (\text{definition } \theta_{\mathcal{L}}) \\
&\iff \quad \forall a \in \mathcal{A} \, \forall s' \in S \colon s \xrightarrow{a}_{\mathcal{L}} s' \Rightarrow \exists t' \in S \colon t \xrightarrow{a}_{\mathcal{L}} t' \wedge R(s',t') \quad (R \text{ is an equivalence relation})
\end{aligned}
$$

We use the logical equivalence from left to right in proving that a bisimulation for $\mathcal{F}(\mathcal{L})$ is a bisimulation equivalence, and the logical equivalence the other way around in proving that a bisimulation equivalence is a *FuTS* bisimulation. □

With appeal to the correspondence result, Theorem 5, we retrieve that strong bisimulation and behavioral equivalence coincide.

4.2   Continuous-time Markov chains

As a first, basic example of a quantitative semantic model we consider continuous-time Markov chains (*CTMC*) and the notion of lumpability. In its purest form, a *CTMC* does not involve actions. It can be viewed as connecting a state to a number of other states while weighing the connection with a real

number, viz. the rate of the negative exponential distribution used to represent the time associated with the transition. As for our treatment of *LTS* we need to restrict to image-finiteness here too, which amounts to finite branching.

Definition 8 (cf. [4])**.**

  (a) *A CTMC is a pair* $\mathcal{C} = (S, \to_{\mathcal{C}})$ *where S is a set of states, and* $\to_{\mathcal{C}} \subseteq S \times \mathbb{R}_{\geqslant 0} \times S$ *is the transition relation. Define* $\mathbf{R}(s,s') = \sum \{\, \lambda \mid s \xrightarrow{\lambda}_{\mathcal{C}} s' \,\}$ *and* $\mathbf{R}(s,C) = \sum \{\, \mathbf{R}(s,s') \mid s' \in C \,\}$.

  (b) *An equivalence relation* $R \subseteq S \times S$ *is called a lumping relation for the CTMC* $\mathcal{C} = (S, \to_{\mathcal{C}})$ *if, for all* $s,t \in S$, *such that* $R(s,t)$ *it holds that* $\mathbf{R}(s,C) = \mathbf{R}(t,C)$ *for every equivalence class C of R.*

  (c) *Two states* $s,t \in S$ *in a CTMC* $\mathcal{C} = (S, \to_{\mathcal{C}})$ *are called lumping equivalent, if there exists a lumping relation R for* $\mathcal{C}$ *such that* $R(s,t)$.

A *CTMC* $\mathcal{C} = (S, \to_{\mathcal{C}})$ induces a simple *FuTS* $\mathcal{F}(\mathcal{C}) = (S, \theta_{\mathcal{C}})$ over the label set $\Delta = \{\delta\}$ and the semiring $\mathbb{R}_{\geqslant 0}$, if we define $\theta_{\mathcal{C}} \colon S \to \mathcal{FS}(S, \mathbb{R}_{\geqslant 0})^{\Delta}$ by

$$\theta_{\mathcal{C}}(s)(\delta)(s') = \sum \{\, \lambda \mid s \xrightarrow{\lambda}_{\mathcal{C}} s' \,\}$$

for $s,s' \in S$. Here, Δ is a dummy set to help *CTMC* fit in the format of *FuTS*, cf. [6, 11]; conventionally, the label δ signifies delay.

Theorem 9. *Let* $\mathcal{C} = (S, \to_{\mathcal{C}})$ *be a CTMC and* $R \subseteq S \times S$ *an equivalence relation. Then it holds that R is a lumping iff R is a FuTS bisimulation for* $\mathcal{F}(\mathcal{C})$.

Proof. Also here the proof mainly consists of unfolding the various definitions. We have, for $s,t \in S$,

$\mathcal{LT}(R, \mathbb{R}_{\geqslant 0})^{\Delta}(s,t)$

$\qquad \Longleftrightarrow \quad \forall C \in S/R \colon \sum_{u \in C} \theta_{\mathcal{C}}(s)(\delta)(u) = \sum_{u \in C} \theta_{\mathcal{C}}(t)(\delta)(u) \qquad\qquad$ (definition $\mathcal{LT}(R, \mathbb{R}_{\geqslant 0})^{\Delta}$)

$\qquad \Longleftrightarrow \quad \forall C \in S/R \colon \sum \{\, \lambda \mid s \xrightarrow{\lambda}_{\mathcal{C}} u, u \in C \,\} = \sum \{\, \mu \mid t \xrightarrow{\mu}_{\mathcal{C}} u, u \in C \,\} \quad$ (definition $\theta_{\mathcal{C}}$)

$\qquad \Longleftrightarrow \quad \forall C \in S/R \colon \mathbf{R}(s,C) = \mathbf{R}(t,C) \qquad\qquad\qquad\qquad\qquad\qquad$ (definition $\mathbf{R}$)

Combing this logical equivalence with the respective definitions of lumping equivalence and bisimulation yields the result. $\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\square$

For *CTMC*, from Theorem 5 and elimination of the degenerated exponentiation with the singleton set Δ, we obtain that lumping equivalence coincides with behaviour equivalence of $\mathcal{FS}(\cdot, \mathbb{R}_{\geqslant 0})$.

4.3   Interactive Markov chains

Interactive Markov chains (*IMC*) were proposed in [15] as a reconciliation of *LTS* and *CTMC*. Because of this two-dimensionality *IMC* constitute a prime example of a combined *FuTS*. The definition of an *IMC* below is taken from [16].

Definition 10. *Fix a set $\mathcal{A}$ of actions.*

(a) *An interactive Markov chain (IMC) over $\mathcal{A}$ is a triple $\mathfrak{I} = (S, \to_{\mathfrak{I}}, \Rightarrow_{\mathfrak{I}})$ where S is a set of states, $\to_{\mathfrak{I}} \subseteq S \times \mathcal{A} \times S$ is the interactive transition relation, and $\Rightarrow_{\mathfrak{I}} \subseteq S \times \mathbb{R}_{\geqslant 0} \times S$ is the Markovian transition relation.*

(b) *For states $s, s' \in S$, action $a \in \mathcal{A}$, and a subset of states $C \subseteq S$, define $\mathbf{T}(s, a, C) \Leftrightarrow s \xrightarrow{a}_{\mathfrak{I}} \bar{s}$ for some state $\bar{s} \in C$. Moreover, define $\mathbf{R}(s, s') = \sum \{ \lambda \mid s \xRightarrow{\lambda}_{\mathfrak{I}} s' \}$ and $\mathbf{R}(s, C) = \sum \{ \mathbf{R}(s, s') \mid s' \in C \}$.*

(c) *An equivalence relation $R \subseteq S \times S$ is called a bisimulation relation for the IMC $\mathfrak{I} = (S, \to_{\mathfrak{I}}, \Rightarrow_{\mathfrak{I}})$ if for all states $s, t \in S$ and every equivalence class C of R the following holds:*

    (i) $\mathbf{T}(s, a, C) = \mathbf{T}(t, a, C)$, *for all $a \in \mathcal{A}$;*

    (ii) $\mathbf{R}(s, C) = \mathbf{R}(t, C)$.

(c) *Two states $s, t \in S$ in an IMC $\mathfrak{I} = (S, \to_{\mathfrak{I}}, \Rightarrow_{\mathfrak{I}})$ are called bisimilar, if there exists a bisimulation relation R for $\mathfrak{I}$ with $R(s, t)$.*

As for *CTMC* we use the symbol δ to denote delay, and put $\Delta = \{\delta\}$. An *IMC* $\mathfrak{I} = (S, \to_{\mathfrak{I}}, \Rightarrow_{\mathfrak{I}})$ over $\mathcal{A}$ induces a combined *FuTS* $\mathcal{F}(\mathfrak{I}) = (S, \theta_{\mathfrak{I}})$, where $\theta_{\mathfrak{I}} = \langle \theta'_{\mathfrak{I}}, \theta''_{\mathfrak{I}} \rangle$, over the label sets $\mathcal{A}$ and Δ and the semirings $\mathbb{B}$ and $\mathbb{R}_{\geqslant 0}$. We define $\theta'_{\mathfrak{I}} : S \to \mathcal{FS}(S, \mathbb{B})^{\mathcal{A}}$ and $\theta''_{\mathfrak{I}} : S \to \mathcal{FS}(S, \mathbb{R}_{\geqslant 0})^{\Delta}$ by

$$\theta'_{\mathfrak{I}}(s)(a)(s') \Leftrightarrow s \xrightarrow{a}_{\mathfrak{I}} s' \quad \text{and} \quad \theta''_{\mathfrak{I}}(s)(\delta)(s') = \mathbf{R}(s, s')$$

for all $s, s' \in S$, $a \in \mathcal{A}$. Thus, the transition function $\theta'_{\mathfrak{I}}$ is similar to the transition function $\theta_{\mathcal{L}}$ of an *LTS*. The transition function $\theta''_{\mathfrak{I}}$ is similar to the transition function $\theta_{\mathcal{C}}$ of a *CTMC*.

The transition relation of a *FuTS* for an *IMC* is the superposition of those of an *LTS* and a *CTMC*. Therefore, the proof of a correspondence result of standard bisimulation and *FuTS* bisimulation for an *IMC* combines the observations made in the proofs of Theorems 7 and 9.

Theorem 11. *Let $\mathfrak{I} = (S, \to_{\mathfrak{I}}, \Rightarrow_{\mathfrak{I}})$ be an IMC and $R \subseteq S \times S$ an equivalence relation. Then it holds that R is a bisimulation for the IMC $\mathfrak{I}$ iff R is a bisimulation for the FuTS $\mathcal{F}(\mathfrak{I})$.*

Proof. For an equivalence relation R and states $s, t \in S$ such that $R(s, t)$ we have the following logical equivalence:

$\mathcal{LT}(R, \mathbb{B})^{\mathcal{A}} \times \mathcal{LT}(R, \mathbb{R}_{\geqslant 0})^{\Delta}(\theta_{\mathfrak{I}}(s), \theta_{\mathfrak{I}}(t))$

$\quad \Leftrightarrow \quad \mathcal{LT}(R, \mathbb{B})^{\mathcal{A}} \times \mathcal{LT}(R, \mathbb{R}_{\geqslant 0})^{\Delta}(\langle \theta'_{\mathfrak{I}}(s), \theta''_{\mathfrak{I}}(s) \rangle, \langle \theta'_{\mathfrak{I}}(t), \theta''_{\mathfrak{I}}(t) \rangle)$ (since $\theta_{\mathfrak{I}} = \langle \theta'_{\mathfrak{I}}, \theta''_{\mathfrak{I}} \rangle$)

$\quad \Leftrightarrow \quad \mathcal{LT}(R, \mathbb{B})^{\mathcal{A}}(\theta'_{\mathfrak{I}}(s), \theta'_{\mathfrak{I}}(t)) \wedge \mathcal{LT}(R, \mathbb{R}_{\geqslant 0})^{\Delta}(\theta''_{\mathfrak{I}}(s), \theta''_{\mathfrak{I}}(t))$ (definition of relational product)

$\quad \Leftrightarrow \quad \forall a \in \mathcal{A} \forall C \in S/R \colon \bigvee \{ \theta'_{\mathfrak{I}}(s)(a)(u) \mid u \in C \} = \bigvee \{ \theta'_{\mathfrak{I}}(t)(a)(u) \mid u \in C \} \wedge$

$\qquad \forall C \in S/R \colon \sum \{ \theta''_{\mathfrak{I}}(s)(\delta)(u) \mid u \in C \} = \sum \{ \theta''_{\mathfrak{I}}(t)(\delta)(u) \mid u \in C \}$

$\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad$ (definition of $\mathcal{LT}(R, \mathbb{B})^{\mathcal{A}}$ and $\mathcal{LT}(R, \mathbb{R}_{\geqslant 0})^{\Delta}$)

$\quad \Leftrightarrow \quad \forall a \in \mathcal{A} \forall C \in S/R \colon \exists u \in C \colon \theta'_{\mathfrak{I}}(s)(a)(u) \Leftrightarrow \exists u \in C \colon \theta'_{\mathfrak{I}}(t)(a)(u) \wedge$

$\qquad \forall C \in S/R \colon \sum \{ \mathbf{R}(s, u) \mid u \in C \} = \sum \{ \mathbf{R}(t, u) \mid u \in C \}$ (definition of sum on $\mathbb{B}$ and $\mathbb{R}_{\geqslant 0}$)

$\quad \Leftrightarrow \quad \forall a \in \mathcal{A} \forall C \in S/R \colon \mathbf{T}(s, a, C) = \mathbf{T}(t, a, C) \wedge \forall C \in S/R \colon \mathbf{R}(s, C) = \mathbf{R}(t, C)$

$\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad$ (definition of $\mathbf{T}$ and $\mathbf{R}$)

Thus, if R is a *FuTS* bisimulation for $\mathfrak{I}$, then $R(s,t)$ implies $\mathcal{LT}(R,\mathbb{B})^{\mathcal{A}} \times \mathcal{LT}(R,\mathbb{R}_{\geqslant 0})^{\Delta}(\theta_{\mathfrak{I}}(s), \theta_{\mathfrak{I}}(t))$. Hence, for all $C \in S/R$, we have $\mathbf{T}(s,a,C) = \mathbf{T}(t,a,C)$ for all $a \in \mathcal{A}$, and $\mathbf{R}(s,C) = \mathbf{R}(t,C)$. So, R is an *IMC* bisimulation. Reversely, if R is an *IMC* bisimulation, then $R(s,t)$ implies for all $C \in S/R$, we have $\mathbf{T}(s,a,C) = \mathbf{T}(t,a,C)$ for all $a \in \mathcal{A}$, and $\mathbf{R}(s,C) = \mathbf{R}(t,C)$. Thus, $\mathcal{LT}(R,\mathbb{B})^{\mathcal{A}} \times \mathcal{LT}(R,\mathbb{R}_{\geqslant 0})^{\Delta}(\theta_{\mathfrak{I}}(s), \theta_{\mathfrak{I}}(t))$. So, R is a *FuTS* bisimulation. □

For *IMC* we have another proof of concrete bisimilarity being equal to behavioral equivalence, a result also presented in [24]. However, here we see better how the bisimulation scheme guides the correspondence result for standard bisimulation for *IMC*, on the one hand, and *FuTS* bisimulation, on the other hand.

5 Quantitative automata as nested and general *FuTS*

In this section we show that *FuTS* and their associated notion of bisimulation suit probabilistic automata as well as Markov automata. For the latter fact to prove we need the full generality of Theorem 5.

5.1 Probabilistic automata

As next quantitative semantic model we consider probabilistic automata (*PA*) originating from [29], and the associated notion of strong Segala bisimulation. We follow the set-up presented in [14].

> **Definition 12.** *Fix a set of actions $\mathcal{A}$.*
>
> (a) *A PA over $\mathcal{A}$ is a pair $\mathcal{P} = (S, \to_{\mathcal{P}})$ where S is a set of states, and $\to_{\mathcal{P}} \subseteq S \times \mathcal{A} \times Distr(S)$ is an image-finite transition relation, i.e. the set $\{ \pi \mid s \xrightarrow{a}_{\mathcal{P}} \pi \}$ is finite, for any state $s \in S$, and action $a \in \mathcal{A}$.*
>
> (b) *An equivalence relation $R \subseteq S \times S$ is called a bisimulation relation for the PA $\mathcal{P} = (S, \to_{\mathcal{P}})$ if, for all $s,t \in S$, $a \in \mathcal{A}$, $\pi \in Distr(S)$ such that $R(s,t)$ and $s \xrightarrow{a}_{\mathcal{P}} \pi$, there exists $\rho \in Distr(S)$ such that $t \xrightarrow{a}_{\mathcal{P}} \rho$ and $\pi[C] = \rho[C]$, for every equivalence class C of R.*
>
> (c) *Two states $s,t \in S$ in a PA $\mathcal{P} = (S, \to_{\mathcal{P}})$ are called probabilistically bisimilar, if there exists a bisimulation relation R for $\mathcal{P}$ such that $R(s,t)$.*

A *PA* $\mathcal{P} = (S, \to_{\mathcal{P}})$ over $\mathcal{A}$ induces a nested *FuTS* $\mathcal{F}(\mathcal{P}) = (S, \theta_{\mathcal{P}})$ with label set $\mathcal{A}$ and semirings $\mathbb{R}_{\geqslant 0}$ and $\mathbb{B}$. We define $\theta_{\mathcal{P}} : S \to \mathcal{FS}(\mathcal{FS}(S, \mathbb{R}_{\geqslant 0}), \mathbb{B})^{\mathcal{A}}$ by

$$\theta_{\mathcal{P}}(s)(a)(\varphi) \iff s \xrightarrow{a}_{\mathcal{P}} \varphi$$

for all $s \in S$, $a \in \mathcal{A}$, $\varphi : S \to \mathbb{R}_{\geqslant 0}$. Note the nesting of $\mathcal{FS}(\cdot, \mathbb{R}_{\geqslant 0})$ and $\mathcal{FS}(\cdot, \mathbb{B})$. Also note that, if $\theta_{\mathcal{P}}(s)(a)(\varphi) = \mathbf{true}$ then φ is in fact a probability distribution, since the probabilistic transition relation connects states to probability distributions only. Finally note, if $\theta_{\mathcal{P}}(s)(a)(\varphi) = \mathbf{false}$ for all φ, then $\mathcal{P}$ admits no *a*-transition for *s*.

The proof of the correspondence result for *PA* is along the same lines as we have seen previously. However, now the equivalence relation R needs to be lifted twice: to the level of $\mathcal{FS}(S, \mathbb{R}_{\geqslant 0})$ first, and to the level of $\mathcal{FS}(\mathcal{FS}(S, \mathbb{R}_{\geqslant 0}), \mathbb{B})$ next.

Theorem 13. *Let $\mathcal{P} = (S, \rightarrow_{\mathcal{P}})$ be a PA and $R \subseteq S \times S$ an equivalence relation. Then it holds that R is a probabilistic bisimulation iff R is a FuTS bisimulation for $\mathcal{F}(\mathcal{P})$.*

Proof. For an equivalence relation $R \subseteq S \times S$ and states $s, t \in S$ such that $R(s, t)$ we have the following:

$$\mathcal{LT}(\mathcal{LT}(R, \mathbb{R}_{\geqslant 0}), \mathbb{B})^{\mathcal{A}}(\theta_{\mathcal{P}}(s), \theta_{\mathcal{P}}(t))$$

$$\Leftrightarrow \quad \forall a \in \mathcal{A} \, \forall \Gamma \in \mathcal{FS}(S, \mathbb{R}_{\geqslant 0}) / \mathcal{LT}(R, \mathbb{R}_{\geqslant 0}) \colon \theta_{\mathcal{P}}(s)(a)(\Gamma) \Leftrightarrow \theta_{\mathcal{P}}(t)(a)(\Gamma)$$
$$\text{(definition } \mathcal{LT}(\mathcal{LT}(R, \mathbb{R}_{\geqslant 0}), \mathbb{B})^{\mathcal{A}})$$

$$\Leftrightarrow \quad \forall a \in \mathcal{A} \, \forall \Gamma \in \mathcal{FS}(S, \mathbb{R}_{\geqslant 0}) / \mathcal{LT}(R, \mathbb{R}_{\geqslant 0}) \colon$$
$$\exists \varphi \in \Gamma \colon \theta_{\mathcal{P}}(s)(\varphi) = \textbf{true} \Leftrightarrow \exists \psi \in \Gamma \colon \theta_{\mathcal{P}}(t)(\psi) = \textbf{true} \qquad \text{(ring structure } \mathbb{B})$$

$$\Leftrightarrow \quad \forall a \in \mathcal{A} \, \forall \Gamma \in \mathcal{FS}(S, \mathbb{R}_{\geqslant 0}) / \mathcal{LT}(R, \mathbb{R}_{\geqslant 0}) \colon \exists \varphi \in \Gamma \colon s \xrightarrow{a}_{\mathcal{P}} \varphi \Leftrightarrow \exists \psi \in \Gamma \colon t \xrightarrow{a}_{\mathcal{P}} \psi$$

$$\Leftrightarrow \quad \forall a \in \mathcal{A} \, \forall \pi \in Distr(S) \colon s \xrightarrow{a}_{\mathcal{P}} \pi \Rightarrow \exists \rho \in Distr(s) \colon t \xrightarrow{a}_{\mathcal{P}} \rho \wedge \forall C \in S/R \colon \pi[C] = \rho[C]$$
$$\text{(symmetry of } R, \ s \xrightarrow{a}_{\mathcal{P}} \varphi \text{ implies } \varphi \in Distr(S))$$

As before, combination of the above equivalence and the respective definitions of *PA* bisimulation for $\mathcal{P}$ and *FuTS* bisimulation for $\mathcal{F}(\mathcal{P})$ yields the result. $\qquad\square$

Thus for *PA* we obtain, via Theorem 5, coalgebraic underpinning using *FuTS* as an intermediate model.

5.2   Markov automata

Markov automata (*MA*) are a relative recent example of a quantitative semantical model [13, 12, 33]. It brings together non-deterministic and probabilistic choice, and stochastic delay.

Definition 14. *Fix a set of actions $\mathcal{A}$.*

(a) *A Markov automaton (MA) over $\mathcal{A}$ is a triple $\mathcal{M} = (S, \rightarrow_{\mathcal{M}}, \Rightarrow_{\mathcal{M}})$ where S is a set of states, $\rightarrow_{\mathcal{M}} \subseteq S \times \mathcal{A} \times Distr(S)$ is the immediate transition relation, and $\Rightarrow_{\mathcal{M}} \subseteq S \times \mathbb{R}_{\geqslant 0} \times S$ is the timed transition relation.*

(b) *For states $s, s' \in S$, action $a \in \mathcal{A}$, and a set of distributions $\Gamma \subseteq Distr(S)$, define $\textbf{T}(s, a, \Gamma) \Leftrightarrow s \xrightarrow{a}_{\mathcal{M}} \pi$ for some distribution $\pi \in \Gamma$. Moreover, define $\textbf{R}(s, s') = \sum \{ \lambda \mid s \xrightarrow{\lambda}_{\mathcal{M}} s' \}$, and, for a set of states C, $\textbf{R}(s, C) = \sum \{ \textbf{R}(s, s') \mid s' \in C \}$.*

(c) *An equivalence relation $R \subseteq S \times S$ is called a bisimulation relation for the MA $\mathcal{M} = (S, \rightarrow_{\mathcal{M}}, \Rightarrow_{\mathcal{M}})$ if, for all states $s, t \in S$ the following holds:*

    (i) *$\textbf{T}(s, a, \Gamma) = \textbf{T}(t, a, \Gamma)$, for all $a \in \mathcal{A}$, and $\Gamma \in Distr(S)/\mathcal{LT}(R, \mathbb{R}_{\geqslant 0})_{Distr(S)}$;*

    (ii) *$\textbf{R}(s, C) = \textbf{R}(t, C)$, for all $C \in S/R$.*

where $\mathcal{LT}(R, \mathbb{R}_{\geqslant 0})_{Distr(S)}$ is the lifting of R to $Distr(S)$; in the sequel we will often omit the subscript $_{Distr(S)}$ for the sake of readability.

(d) *Two states $s, t \in S$ in an MA $\mathcal{M} = (S, \rightarrow_{\mathcal{M}}, \Rightarrow_{\mathcal{M}})$ are called bisimilar, if there exists a bisimulation relation R for $\mathcal{M}$ with $R(s, t)$.*

In part (c), the definition of a bisimulation relation for an *MA*, we encounter the use of two kinds of equivalence relations. For the timed behaviour the comparison is at the level of states involving equivalence classes from S/R. For the immediate behaviour, though, the comparison is at the level of distributions of states involving equivalence classes from $Distr(S)/\mathcal{LT}(R,\mathbb{R}_{\geq 0})_{Distr(S)}$, where $\mathcal{LT}(R,\mathbb{R}_{\geq 0})_{Distr(S)}$ denotes the lifting of R with respect to $Distr(S)$, see Section 2. Note, for *PA*, because of the direct definition used, the application of the lifting operator is left implicit.

Again, distinguish the symbol δ to denote delay, and put $\Delta = \{\delta\}$. An *MA* $\mathcal{M} = (S, \rightarrow_{\mathcal{M}}, \Rightarrow_{\mathcal{M}})$ over $\mathcal{A}$ induces a general *FuTS* $\mathcal{F}(\mathcal{M}) = (S, \theta_{\mathcal{M}})$, with $\theta_{\mathcal{M}} = \langle \theta'_{\mathcal{M}}, \theta''_{\mathcal{M}} \rangle$, over the label sets $\mathcal{A}$ and Δ and the sequences of semirings $\mathbb{R}_{\geq 0}$, $\mathbb{B}$ and $\mathbb{R}_{\geq 0}$, if we define $\theta'_{\mathcal{M}} : S \rightarrow \mathcal{FS}(\mathcal{FS}(S, \mathbb{R}_{\geq 0}), \mathbb{B})^{\mathcal{A}}$ and $\theta''_{\mathcal{M}} : S \rightarrow \mathcal{FS}(S, \mathbb{R}_{\geq 0})^{\Delta}$ by

$$\theta'_{\mathcal{M}}(s)(a)(\varphi) \Leftrightarrow s \xrightarrow{a}_{\mathcal{M}} \varphi \quad \text{and} \quad \theta''_{\mathcal{M}}(s)(\delta)(s') = \mathbf{R}(s, s')$$

for $s, s' \in S$, $a \in \mathcal{A}$, $\varphi \in \mathcal{FS}(S, \mathbb{R}_{\geq 0})$. Note, the *FuTS* $\mathcal{F}(\mathcal{M})$ is a combination of a nested *FuTS* representing the immediate behaviour of $\mathcal{M}$ and a simple *FuTS* representing the timed behaviour of $\mathcal{M}$.

Theorem 15. *Let $\mathcal{M} = (S, \rightarrow_{\mathcal{M}}, \Rightarrow_{\mathcal{M}})$ be a MA and $R \subseteq S \times S$ an equivalence relation. Then it holds that R is an MA bisimulation for $\mathcal{M}$ iff R is a FuTS bisimulation for $\mathcal{F}(\mathcal{M})$.*

Proof. For a binary relation $R \subseteq S \times S$ and states $s, t \in S$ such that $R(s, t)$ we have the following logical equivalence:

$$\left(\mathcal{LT}(\mathcal{LT}(R, \mathbb{R}_{\geq 0}), \mathbb{B})^{\mathcal{A}} \times \mathcal{LT}(R, \mathbb{R}_{\geq 0})^{\Delta} \right)(\theta_{\mathcal{M}}(s), \theta_{\mathcal{M}}(t))$$

$$\Leftrightarrow \quad \mathcal{LT}(\mathcal{LT}(R, \mathbb{R}_{\geq 0}), \mathbb{B})^{\mathcal{A}}(\theta'_{\mathcal{M}}(s), \theta'_{\mathcal{M}}(t)) \wedge \mathcal{LT}(R, \mathbb{R}_{\geq 0})^{\Delta}(\theta''_{\mathcal{M}}(s), \theta''_{\mathcal{M}}(t))$$

$$(\theta_{\mathcal{M}} = \langle \theta'_{\mathcal{M}}, \theta''_{\mathcal{M}} \rangle, \text{ definition relational product})$$

$$\Leftrightarrow \quad \forall a \in \mathcal{A} \forall \pi \in Distr(S):$$

$$s \xrightarrow{a}_{\mathcal{P}} \pi \Rightarrow \exists \rho \in Distr(s): t \xrightarrow{a}_{\mathcal{P}} \rho \wedge \forall C \in S/R: \pi[C] = \rho[C] \wedge$$

$$\forall C \in S/R: \sum \{ \lambda \mid s \xrightarrow{\lambda}_{\mathcal{C}} u, u \in C \} = \sum \{ \mu \mid t \xrightarrow{\mu}_{\mathcal{C}} u, u \in C \}$$

$$(\text{see the proofs of Theorems 13 and 9})$$

$$\Leftrightarrow \quad \forall a \in \mathcal{A} \forall \Gamma \in Distr(S)/\mathcal{LT}(R): \mathbf{T}(s, a, \Gamma) = \mathbf{T}(t, a, \Gamma) \wedge$$

$$\forall C \in S/R: \mathbf{R}(s, C) = \mathbf{R}(t, C)$$

$$(\text{symmetry } R, \text{ definition } \mathbf{T} \text{ and } \mathbf{R})$$

As previously, combination of the above equivalence and the respective definitions of *MA* bisimulation for $\mathcal{M}$ and *FuTS* bisimulation for $\mathcal{F}(\mathcal{M})$ yields the result. $\qquad \square$

Markov automata feature, so to speak, simultaneously non-deterministic branching followed by probabilistic branching on the one hand, vs. stochastic branching on the other hand. This is reflected by the corresponding *FuTS* being of type $\mathcal{FS}(\mathcal{FS}(\cdot, \mathbb{R}_{\geq 0}), \mathbb{B})^{\mathcal{A}} \times \mathcal{FS}(\cdot, \mathbb{R}_{\geq 0})^{\Delta}$. Again, by the correspondence theorem, behavioral equivalence of the type functor captures exactly the concrete notion of bisimulation.

6   Concluding remarks

We contributed to the work of providing uniform techniques for modeling quantitative process languages. The concept of a *FuTS*, originating from [10], provides a compact way to assign quantities to states, or other entities, by means of continuations, leading to clean and concise descriptions of *SPC*, cf. [11].

   In the current paper, we extended the results we presented in [24]. Here, we associate a type to a *FuTS*, and we propose a systematic way, directed by the type, of lifting an equivalence relation from the level of states to the level of the continuations involved. The scheme allows types with products and arbitrary nesting. The induced notion of *FuTS* bisimulation coincides with behavioral equivalence, a coalgebraic notion of identification associated with the **Set** functor implied by the type. In particular, the correspondence result proved in [24] now extends to nested *FuTS* and related functors. Various forms of quantitative transition systems are shown to be amenable to a representation as a *FuTS* including *IMC*, *PA* and *MA*. Moreover, the concrete notion of bisimulation of these quantitative transition systems is shown to coincide with the notion of *FuTS* bisimulation as given by our scheme, and hence with behavioral equivalence.

   The main restriction of *FuTS* is its being based on finitely supported functions. As a consequence, image-finiteness is inherent to our treatment. Still the scope of application is broad. Apart from the examples discussed here and in our earlier work [24, 22] we have been able to model discrete real-time with *FuTS* as well [23]. However, replacing the construct $\mathcal{FS}(\,\cdot\,,\mathcal{R})$ by the Giry monad of measurable functions [27] may be an option: on the one hand, finiteness is traded for a restricted form of infinite, on the other hand, summation is exchanged for integration. However, in the continuous setting, e.g. for hybrid systems, broadly accepted semantical models that parallel *LTS* seem to be missing.

   In this paper we focused on strong notions of bisimilarity. Weak notions prove difficult to handle, e.g. for probabilistic weak bisimulation [12] or for probabilistic branching bisimulation [2], see also [31]. It would be interesting to investigate whether the more abstract view based on *FuTS* and a categorical approach, in the line of [25, 8] could be of help.

References

[1]   J. Adámek & H.-E. Porst (2004): *On tree coalgebras and coalgebra presentations*. Theoretical Computer Science 311, pp. 257–283, doi:10.1016/S0304-3975(03)00378-5.

[2]   S. Andova, S. Georgievska & N. Trčka (2012): *Branching bisimulation congruence for probabilistic systems*. Theoretical Computer Science 413, pp. 58–72, doi:10.1016/j.tcs.2011.07.020.

[3]   C. Baier (1998): *On Algorithmic Verification Methods for Probabilistic Systems*. University of Mannheim. Hablitations Thesis.

[4]   C. Baier, H. Hermanns, J.-P. Katoen & V. Wolf (2006): *Bisimulation and Simulation Relations for Markov Chains*. Electronic Notes in Theoretical Computer Science 162, pp. 73–78, doi:10.1016/j.entcs.2005.12.078.

[5]   F. Bartels, A. Sokolova & E.P. de Vink (2004): *A hierarchy of probabilistic system types*. Theoretical Computer Science 327, pp. 3–22, doi:10.1016/j.tcs.2004.07.019.

[6] M. Bernardo, R. De Nicola & M. Loreti (2013): *A uniform framework for modeling nondeterministic, probabilistic, stochastic, or mixed processes and their behavioral equivalences*. Information and Computation 225, pp. 29–82, doi:10.1016/j.ic.2013.02.004.

[7] F. Bonchi, M. Bonsangue, M. Boreale, J. Rutten & A. Silva (2012): *A coalgebraic perspective on linear weighted automata*. Information and Computation 211, pp. 77–105, doi:10.1016/j.ic.2011.12.002.

[8] T. Brengos, M. Miculan & M. Peressotti (2014): *Behavioural equivalences for coalgebras with unobservable moves*. CoRR abs/1411.0090. Available at `http://arxiv.org/abs/1411.0090`.

[9] S. Crafa & F. Ranzato (2011): *A Spectrum of Behavioral Relations over LTSs on Probability Distributions*. In J.-.P Katoen & B. König, editors: *Proc. CONCUR 2011*, LNCS 6901, pp. 124–139, doi:10.1007/978-3-642-23217-6_9.

[10] R. De Nicola, D. Latella, M. Loreti & M. Massink (2009): *Rate-based Transition Systems for Stochastic Process Calculi*. In S. Albers et al., editor: *Proc. ICALP 2009, Part II*, LNCS 5556, pp. 435–446, doi:10.1007/978-3-642-02930-1_36.

[11] R. De Nicola, D. Latella, M. Loreti & M. Massink (2013): *A Uniform Definition of Stochastic Process Calculi*. ACM Computing Surveys 46, pp. 5:1–5:35, doi:10.1145/2522968.2522973.

[12] C. Eisentraut, H. Hermanns & Lijun Zhang (2010): *Concurrency and Composition in a Stochastic World*. In P. Gastin & F. Laroussinie, editors: *Proc. CONCUR 2010*, LNCS 6269, pp. 21–39, doi:10.1007/978-3-642-15375-4_3.

[13] C. Eisentraut, H. Hermanns & Lijun Zhang (2010): *On Probabilistic Automata in Continuous Time*. In: *Proc. LICS, Edinburgh*, IEEE Computer Society, pp. 342–351.

[14] M. Hennessy (2012): *Exploring probabilistic bisimulations, part I*. Formal Aspects of Computing 24, pp. 749–768, doi:10.1007/s00165-012-0242-7.

[15] H. Hermanns (2002): *Interactive Markov Chains: The Quest for Quantified Quality*. LNCS 2428, doi:10.1007/3-540-45804-2.

[16] H. Hermanns & J.-P. Katoen (20010): *The How and Why of Interactive Markov Chains*. In F.S. de Boer, M.M. Bonsangue, S. Hallerstede & M. Leuschel, editors: *Proc. FMCO 2009*, LNCS 6286, pp. 311–337, doi:10.1007/978-3-642-17071-3_16.

[17] J. Hillston (1996): *A Compositional Approach to Performance Modelling*. Distinguished Dissertations in Computer Science 12, Cambridge University Press, doi:10.1017/CBO9780511569951.

[18] B. Klin (2009): *Structural Operational Semantics for Weighted Transition Systems*. In J. Palsberg, editor: Semantics and Algebraic Specification, LNCS 5700, pp. 121–139, doi:10.1007/978-3-642-04164-8_7.

[19] B. Klin & V. Sassone (2008): *Structural Operational Semantics for Stochastic Process Calculi*. In R.M. Amadio, editor: *Proc. FoSSaCS 2008*, LNCS 4962, pp. 428–442, doi:10.1007/978-3-540-78499-9_30.

[20] A. Kurz (2000): *Logics for coalgebras and applications to computer science*. Ph.D. thesis, LMU München.

[21] K.G. Larsen & A. Skou (1991): *Bisimulation through Probabilistic Testing*. Information and Computation 94, pp. 1–28, doi:10.1016/0890-5401(91)90030-6.

[22] D. Latella, M. Massink & E.P. de Vink (2013): *Coalgebraic Bisimulation of FuTS*. Technical Report TR 09, ASCENS: Autonomic Service-Component Ensembles (EU Proj. 257414).

[23] D. Latella, M. Massink & E.P. de Vink: *Bisimulation of Labeled State-to-Function Transition Systems Coalgebraically*. Submitted.

[24] D. Latella, M. Massink & E.P. de Vink (2012): *Bisimulation of Labeled State-to-Function Transition Systems of Stochastic Process Languages*. In U. Golas & T. Soboll, editors: *Proc. ACCAT 2012*, EPTSC 93, pp. 23–43, doi:10.4204/EPTCS.93.2.

[25] M. Miculan & M. Peressotti (2013): *Weak bisimulations for labelled transition systems weighted over semirings*. CoRR abs/1310.4106. Available at `http://arxiv.org/abs/1310.4106`.

[26] M. Miculan & M. Peressotti (2014): *GSOS for non-deterministic processes with quantitative aspects*. In N. Bertrand & L. Bortolussi, editors: *Proc. QAPL 2014*, EPTCS 154, pp. 17–33, doi:10.4204/EPTCS.154.2.

[27] P. Panangaden (2009): *Labelled Markov Processes*. Imperial College Press, doi:10.1142/9781848162891.

[28] J.J.M.M. Rutten (2000): *Universal coalgebra: a theory of systems*. Theoretical Computer Science 249, pp. 3–80, doi:10.1016/S0304-3975(00)00056-6.

[29] R. Segala & N.A. Lynch (1995): *Probabilistic Simulations for Probabilistic Processes*. Nordic Journal of Computing 2, pp. 250–273.

[30] A. Sokolova (2011): *Probabilistic systems coalgebraically: A survey*. Theoretical Computer Science 412, pp. 5095–5110, doi:10.1016/j.tcs.2011.05.008.

[31] A. Sokolova, E.P. de Vink & H. Woracek (2009): *Coalgebraic Weak Bisimulation for Action-Type Systems*. Scientific Annals of Computer Science 19, pp. 93–144.

[32] S. Staton (2011): *Relating coalgebraic notions of bisimulation*. Logical Methods in Computer Science 7, pp. 1–21, doi:10.2168/LMCS-7(1:13)2011.

[33] M. Timmer, J.-P. Katoen, J. van de Pol & M. Stoelinga (2012): *Efficient Modelling and Generation of Markov Automata*. In M. Koutny & I. Ulidowski, editors: *Proc. CONCUR 2012*, LNCS 7454, pp. 364–379, doi:10.1007/978-3-642-32940-1_26.