Structural operational semantics for non-deterministic processes with quantitative aspects*

Marino Miculan
marino.miculan@uniud.it

Marco Peressotti
marco.peressotti@uniud.it

Laboratory of Models and Applications of Distributed Systems
Department of Mathematics and Computer Science
University of Udine, Italy

Abstract

Recently, unifying theories for processes combining non-determinism with quantitative aspects (such as probabilistic or stochastically timed executions) have been proposed with the aim of providing general results and tools. This paper provides two contributions in this respect. First, we present a general GSOS specification format and a corresponding notion of bisimulation for non-deterministic processes with quantitative aspects. These specifications define labelled transition systems according to the ULTraS model, an extension of the usual LTSs where the transition relation associates any source state and transition label with *state reachability weight functions* (like, e.g., probability distributions). This format, hence called *Weight Function GSOS* (WF-GSOS), covers many known systems and their bisimulations (e.g. PEPA, TIPP, PCSP) and GSOS formats (e.g. GSOS, Weighted GSOS, Segala-GSOS).

The second contribution is a characterization of these systems as coalgebras of a class of functors, parametric in the weight structure. This result allows us to prove *soundness* and *completeness* of the WF-GSOS specification format, and that bisimilarities induced by these specifications are always congruences.

1 Introduction

Process calculi and labelled transition systems have proved very successful for modelling and analysing concurrent, non-deterministic systems. This success has led to many extensions dealing with quantitative aspects, whose transition relations are endowed with further information like probability rates or stochastic rates; see [6, 5, 16, 21, 25] among others. These calculi are very effective in modelling and analysing quantitative aspects, like performance analysis of computer networks, model checking of time-critical systems, simulation of biological systems, probabilistic analysis of security and safety properties, etc.

Each of these calculi is tailored to a specific quantitative aspect and for each of them we have to develop a quite complex theory almost from scratch. This is a daunting and error-prone task, as it embraces the definition of syntax, semantics, transition rules, various behavioural equivalences, logics, proof systems; the proof of important properties like congruence of behavioural equivalences; the development of algorithms and tools for simulations, model checking, etc. This situation would naturally benefit from general *frameworks* for LTS with quantitative aspects, i.e., mathematical *meta*models offering general methodologies, results, and tools, which can be

*This work is partially supported by MIUR PRIN project 2010LHT4KM, *CINA*.

uniformly instantiated to a wide range of specific calculi and models. In recent years, some of these theories have been proposed; we mention *Segala systems* [27], *Functional Transition Systems* (FuTS) [23], *weighted labelled transition systems* (WLTSs) [14, 21], and *Uniform Labelled Transition Systems* (ULTraS), introduced by Bernardo, De Nicola and Loreti specifically as "a uniform setting for modelling non-deterministic, probabilistic, stochastic or mixed processes and their behavioural equivalences" [5].

A common feature of most of these meta-models is that their labelled transition relations do not yield simple states (e.g., processes), but some mathematical object representing quantitative information about "how" each state can be reached. In particular, transitions in ULTraS systems have the form $P \xrightarrow{a}\!\!\rhd\ \rho$ where ρ is a *state reachability weight function*, i.e., a function assigning a *weight* to each possible state.[1] By suitably choosing the set of weights, and how these functions can be combined, we can recover ordinary non-deterministic LTSs, probabilistic transition systems, stochastic transition systems, etc. As convincingly argued in [5], the use of weight functions in place of plain processes simplifies the combination of non-determinism with quantitative aspects, like in the case of EMPA or PEPA. Moreover, it paves the way for general definitions and results, an important example being the notion of $\mathcal{M}$-bisimulation [5].

Albeit quite effective, these meta-models are at their dawn, with many results and techniques still to be developed. An important example of these missing notions is a *specification format*, like the well-known GSOS, ntyft/ntyxt and ntree formats for non-deterministic labelled transition systems. These formats are very useful in practice, because they can be used for ensuring important properties of the system; in particular, the bisimulations induced by systems in these formats is guaranteed to be a congruence (which is crucial for compositional reasoning). From a more foundational point of view, these frameworks would benefit from a categorical characterization in the theory of coalgebras and bialgebras: this would allow a cross-fertilizing exchange of definitions, notions and techniques with similar contexts and theories.

In this paper, we provide two main contributions in this respect. First, we present a GSOS-style format, called *Weight Function GSOS* (WF-GSOS), for the specifications of non-deterministic systems with quantitative aspects. The judgement derived by rules in this style is of the form $P \xrightarrow{a}\!\!\rhd\ \psi$, where P is a process and ψ is a *weight function term*. These terms describe weight functions by means of an *interpretation*; hence, a specification given in this format defines a ULTraS. By choosing the set of weights, the language of weight function terms and their interpretation, we can readily capture many quantitative notions (probabilistic, stochastic, etc.), and different kinds of non-deterministic interactions, covering models like PEPA, TIPP, PCSP, EMPA, among others. Moreover, the WF-GSOS format supports a general definition of *(strong) bisimulation*, which can be readily instantiated to the various specific systems.

The second contribution is more fundamental. We provide a general categorical presentation of these non-deterministic systems with quantitative aspects. Namely, we prove that ULTraS systems are in one-to-one correspondence with coalgebras of a precise class of functors, parametric on the underlying weight structure. Using this characterization we define the abstract notion of *WF-GSOS distributive law* (i.e. a natural transformation of a specific shape) for these functors. We show that each WF-GSOS specification yields such a distributive law (i.e., the format is sound); taking advantage of Turi-Plotkin's bialgebraic framework, this implies that the bisimulation induced by a WF-GSOS is always a congruence, thus allowing for compositional reasoning in quantitative settings. Additionally, we extend the results we presented in [24] proving that the WF-GSOS format is also *complete*: every abstract WF-GSOS distributive law for ULTraSs can be described by means of some WF-GSOS specification.

[1] The reader aware of advanced process calculi will be not baffled by the fact that targets are not processes. Well known previous examples are the LTS abstractions/concretions for π-calculus, for the applied π-calculus, for the ambient calculus, etc.

The rest of the paper is structured as follows. In Section 2 we recall Uniform Labelled Transition Systems, and their bisimulation. In Section 3 we introduce the *Weight Function SOS* specification format for the syntactic presentation of ULTraSs. In Section 4 we provide some application examples, such as a WF-GSOS specification for PEPA and the translations of Segala-GSOS and WGSOS specifications in the WF-GSOS format. The categorical presentation of ULTraS and WF-GSOS, with the results that the format is sound and complete and bisimilarity is a congruence, are in Section 5. Final remarks, comparison with related work and directions for future work are in Section 6.

2 Uniform Labelled Transition Systems and their bisimulation

In this section we recall and elaborate the definition of ULTraSs, and define the corresponding notion of (coalgebraically derived) bisimulation; finally we compare it with the notion of $\mathcal{M}$-bisimulation presented in [5]. Additional examples are provided in the Appendix. Although we focus on the ULTraS framework, the results and methodologies described in this paper can be ported to similar formats (like FuTS [23]), and more generally to a wide range of systems combining computational aspects in different ways.

2.1 Uniform Labelled Transition Systems

ULTraS are (non-deterministic) labelled transition systems whose transitions lead to *state reachability weight functions*, i.e. functions representing quantitative information about "how" each state can be reached. Examples of weight functions include probability distributions, resource consumption levels, or stochastic rates. In this light, ULTraS can be thought of as a generalization of Segala systems [27], which stratify non-determinism over probability. Following the parallel with Segala systems, ULTraS transitions can be pictured as being composed by two steps:

$$x \xrightarrow{a}\!\!\rhd\; \rho \xrightarrow{w}\!\!\blacktriangleright\; y$$

where the first is a labelled non-deterministic (sub)transition and the second is a weighted one; from this perspective the weight function plays the rôle of the "hidden intermediate state".

Akin to Weighted Labelled Transition Systems (WLTS) [21, 14], weights are drawn from a fixed set endowed with a commutative monoid structure, where the unit is meant to be assigned to disabled transitions (i.e. those yielding unreachable states) and the monoidal addition is used to compositionally weigh sets of transitions given by non-determinism.

Definition 2.1 ($\mathfrak{W}$-ULTraS)**.** *Given a commutative monoid $\mathfrak{W} = (W, +, 0)$, a* ($\mathfrak{W}$-weighted) Uniform Labelled Transition System *is a triple $(X, A, \rightarrow\!\!\rhd)$ where:*

- *X is a set of* states *(processes) called* state space *or* carrier*;*
- *A is a set of* labels *(actions);*
- *$\rightarrow\!\!\rhd \subseteq X \times A \times [X \to W]$ is a* transition relation *where $[X \to W]$ denotes the set of all* weight functions *from X to the carrier of $\mathfrak{W}$.*

Monoidal addition does not play any rôle in the above definition[2] but it is crucial to define the notion of bisimulation and in general how the "merging" of two states (e.g. induced by

[2] Originally, in [5] W is a partial order with bottom. Actually, the order is not crucial to the basic definition of ULTraS as it is only used by some equivalences considered in that paper.

functions between carriers) affects the transition relation. In fact, bisimulations can be thought as inducing "state space refinements that are well-behaved w.r.t. the transition relation". From this perspective, monoidal addition provides an *abstract, uniform and compositional* way to "merge" the outgoing transitions into one: adding their weight; likewise probabilities or stochastic rates are added in probabilistic or stochastic systems.

Because the monoidal structure supports finite addition only[3] we can only merge finitely many transitions. Assuming ULTraSs to have a finite carrier or maps between carriers to define finite pre-images (i.e. $|f^{-1}(y)| \in \mathbb{N}$) is preposterous: since we aim to provide syntactic description of ULTraSs, state spaces may be infinite (cf. initial semantics) and functions may map arbitrary many states to the same image, e.g., their behaviour (cf. bisimulations, final semantics). Therefore, in this paper we shall consider *image finite* ULTraSs only. This is a mild and common assumption (e.g. [21, 4, 7]) and our results readily generalise to transfinite bounds (e.g. to deal with countably-branching systems).

Definition 2.2 (Image finiteness)**.** *Let $\mathfrak{W} = (W, +, 0)$ be a commutative monoid. For a function $\rho : X \to W$ the set $\lfloor\rho\rfloor \triangleq \{x \mid \rho(x) \neq 0\}$ is called* support of ρ *and whenever it is finite ρ is said to be* finitely supported. *The set of finitely supported functions with domain X is denoted by $\mathcal{F}_{\mathfrak{W}}X$. A $\mathfrak{W}$-ULTraS $(X, A, \multimap\!\!\!\rhd)$ is said to be* image finite *iff for any state $x \in X$ and label $a \in A$ the set $\{\rho \mid x \xrightarrow{a}\!\!\rhd\ \rho\}$ is finite and contains only finitely supported weight functions.*

Example 2.3. A weight function $\rho \in \mathcal{F}_{2}X$ (for $2 = (\{\mathtt{tt}, \mathtt{ff}\}, \vee, \mathtt{ff})$) is a predicate describing a finite subset of X. Thus $\mathcal{P}_f X \cong \mathcal{F}_2 X$. Likewise, a function $\rho \in \mathcal{F}_{\mathfrak{N}}X$ (for $\mathfrak{N} = (\mathbb{N}, +, 0)$) assigns to each element of X a multiplicity and hence describes a finite multiset.

Intuitively, elements of $\mathcal{F}_{\mathfrak{W}}X$ can be seen as "generalised multisets". Therefore, it is natural to extend a function $f : X \to Y$ to a function $\mathcal{F}_{\mathfrak{W}}(f) : \mathcal{F}_{\mathfrak{W}}X \to \mathcal{F}_{\mathfrak{W}}Y$ mapping (finitely supported) weight functions over X to (finitely supported) weight functions over Y as follows:

$$\mathcal{F}_{\mathfrak{W}}(f)(\rho) \triangleq \lambda y : Y. \textstyle\sum_{x \in f^{-1}(y)} \rho(x). \tag{1}$$

This definition generalises the extension of a function to the powerset; in fact, $\mathcal{F}_2(f)(\rho) = \lambda y : Y. \bigvee_{x \in f^{-1}(y)} \rho(x)$ describes the subset of Y whose elements are image of some element in the subset of X described by ρ. Henceforth, we shall refer to $\mathcal{F}_{\mathfrak{W}}(f)(\rho)$ as the *action of f on ρ* and denote it by $\rho[f]$, when confusion seems unlikely.

We can now make the idea of "state space maps being well-behaved w.r.t. the transition relation" formal:

Definition 2.4 (ULTraS homomorphism)**.** *Let $(X, A, \multimap\!\!\!\rhd_X)$ and $(Y, A, \multimap\!\!\!\rhd_Y)$ be two image-finite $\mathfrak{W}$-ULTraS. A* homomorphism $f : (\multimap\!\!\!\rhd_X) \to (\multimap\!\!\!\rhd_Y)$ *is a function $f : X \to Y$ between their state spaces such that for any $x \in X$ and $a \in A$:*

$$x \xrightarrow{a}\!\!\rhd_X \rho \iff f(x) \xrightarrow{a}\!\!\rhd_Y \rho[f].$$

Given two homomorphisms $f : (\multimap\!\!\!\rhd_X) \to (\multimap\!\!\!\rhd_Y)$ and $g : (\multimap\!\!\!\rhd_Y) \to (\multimap\!\!\!\rhd_Z)$, the function $g \circ f : X \to Z$ is a homomorphism $g \circ f : (\multimap\!\!\!\rhd_X) \to (\multimap\!\!\!\rhd_Z)$. Homomorphism composition is always defined, it is associative and has identities. In Section 5 we will show that ULTraSs homomorphisms indeed form categories equivalent to categories of coalgebras for a suitable functor. For the time being, consider the degenerate monoid 1 containing exactly its unit and let A be a singleton; then a 1-ULTraS $(X, A, \multimap\!\!\!\rhd_X)$ is just a relation $\multimap\!\!\!\rhd_X \cong R_X$ on X and any homomorphism is exactly a relation homomorphism. In fact, $f : X \to Y$ is a 1-ULTraS homomorphism

[3] Indeed it is possible to assume sums for any family indexed by some set; however, in Section 5 we assume image-finiteness to guarantee the existence of a final coalgebra.

$f : (\rightarrowtriangle_X) \to (\rightarrowtriangle_Y)$ iff $(x, x') \in R_x \iff (f(x), f(x')) \in R_Y$. For A with more than one label we get exactly homomorphisms of labelled relations i.e. LTSs.

2.2 Bisimulation

We present now the definition of bisimulation for ULTraS based on the notion of *kernel bisimulation* (a.k.a. behavioural equivalence) i.e. "a relation which is the kernel of a common compatible refinement of the two[4] systems" [28]. This notion naturally stems from the final semantics approach and, under mild assumptions, coincides with Aczel-Medler's coalgebraic bisimulation, as we will see in Section 5.

Definition 2.5 (Refinement)**.** *Given* $(X, A, \rightarrowtriangle_X)$ *a* refinement *for it is any* $(Y, A, \rightarrowtriangle_Y)$ *such that there exists an homomorphism* $f : (\rightarrowtriangle_X) \to (\rightarrowtriangle_Y)$.

Homomorphisms provide the right notion of refinement. Consider an equivalence relation $R \subseteq X \times X$, R is *stable w.r.t.* $\rightarrowtriangle_X$ if, and only if, its equivalence classes are not split by the transition relation $\rightarrowtriangle_X$, i.e., iff there is a refinement whose carrier is $Y = X/R$. Hence, stability of an equivalence relation corresponds to the canonical projection $\kappa : X \to X/R$ being a ULTraS homomorphism. This observation contains all the ingredients needed to define bisimulations for ULTraSs. Before we formalise this notion let us introduce some accessory notation.

In the following, we will denote the *total weight* of $\rho \in \mathcal{F}_{\mathfrak{W}}X$ by $\llbracket \rho \rrbracket \triangleq \sum_{x \in X} \rho(x)$. The weight ρ assigned to $C \subseteq X$ is the total weight of the restriction $\rho\big|_C$ i.e. $\llbracket \rho\big|_C \rrbracket = \sum_{x \in C} \rho(x)$. Any relation R between two sets X and Y defines a relation $R_{\mathfrak{W}}$ between finitely supported weight functions for X and Y as:

$$(\phi, \psi) \in R_{\mathfrak{W}} \overset{\triangle}{\iff} \forall (C, D) \in R^\star \; \llbracket \phi\big|_C \rrbracket = \llbracket \psi\big|_D \rrbracket$$

where $R^\star \subseteq \mathcal{P}X \times \mathcal{P}Y$ is the *subset closure* of R i.e. smallest relation s.t., for $C \subseteq X$, $D \subseteq Y$:

$$\begin{aligned}(C, D) \in R^\star \iff & (\forall x \in C, \forall y \in Y : (x, y) \in R \Rightarrow y \in D) \wedge \\ & (\forall x \in X, \forall y \in D : (x, y) \in R \Rightarrow x \in C)\end{aligned}$$

Definition 2.6 (Bisimulation)**.** *Let* $(X, A, \rightarrowtriangle_X)$ *and* $(Y, A, \rightarrowtriangle_Y)$ *be two image-finite* $\mathfrak{W}$*-ULTraS. A relation* R *between* X *and* Y *is a* bisimulation *if, and only if, for each pair of states* $x \in X$ *and* $y \in Y$*,* $(x, y) \in R$ *implies that for each label* $a \in A$ *the following hold:*

- *if* $x \xrightarrowtriangle{a}_X \phi$ *then there exists* $y \xrightarrowtriangle{a}_Y \psi$ *s.t.* $(\phi, \psi) \in R_{\mathfrak{W}}$.
- *if* $y \xrightarrowtriangle{a}_Y \psi$ *then there exists* $x \xrightarrowtriangle{a}_X \phi$ *s.t.* $(\phi, \psi) \in R_{\mathfrak{W}}$.

Processes x *and* y *are said to be* bisimilar *if there exists a bisimulation relation* R *such that* $(x, y) \in R$.

As ULTraSs can be seen as stacking non-determinism over other computational behaviour, Definition 2.6 stratifies bisimulation for non-deterministic labelled transition system over bisimulation for systems expressible as labelled transition systems weighted over commutative monoids. In fact, two processes x and y are related by some bisimulation if, and only if, whether one reaches

[4]We present bisimulations as relations between two state spaces instead of considering one system in isolation; we are aware that in the case of ULTraS two systems can be "run in parallel" still the notion of having a common refinement allows for different homomorphisms even when considering a single system and therefore offers greater generality.

a weight function via a non-deterministic labelled transition, the other can reach another function via a transition with the same label, where the two functions are equivalent in the sense that they assign the same total weight to the classes of states in the relation. For instance, in the case of weights being probabilities, functions are considered equivalent only when they agree on the probabilities assigned to each class of states which is precisely the intuition behind probabilistic bisimulation [22]. More examples will be discussed below and in the Appendix.

Constrained ULTraS Sometimes, the ULTraSs induced by a given monoid are too many, and we have to restrict to a subclass. For instance, fully-stochastic systems such as (labelled) CTMCs are a strict subclass of ULTraSs weighted over the monoid of non-negative real numbers $(\mathbb{R}_0^+, +, 0)$, where weights express rates of exponentially distributed continuous time transitions. In the case of fully-stochastic systems, for each label, each state is associated with precisely one weight function. This kind of "deterministic" ULTraSs are called *functional* in [5], because the transition relation is functional, and correspond precisely to WLTSs [21, 14]. These are a well-known family of systems (especially their automata counterpart) and have an established coalgebraic understanding as long as a (coalgebraically derived) notion of *weighted bisimulation* which are shown to subsume several known kinds of systems such as non-deterministic, (fully) stochastic, generative and reactive probabilistic [21]. Moreover, Definition 2.6 coincides with weighted bisimulation on functional ULTraSs/WLTSs over the same monoid [21, Def. 4]; hence Definition 2.6 covers every system expressible in the framework of WLTS. (cf. Appendix A).

Proposition 2.7. *Let $\mathfrak{W}$ be a commutative monoid and $(X, A, \rightarrowtriangle_X)$, $(Y, A, \rightarrowtriangle_Y)$ be $\mathfrak{W}$-LTSs seen as a functional $\mathfrak{W}$-ULTraSs. Every bisimulation relation between them is a $\mathfrak{W}$-weighted bisimulation and vice versa.*

Proof (Omitted). See Appendix A. □

Another constraint arises in the case of probabilistic systems, i.e., weight functions are probability distributions. Since addition is not a closed operation in the unit interval $[0, 1]$, there is no monoid $\mathfrak{W}$ such that every weight function on it is also a probability distribution. Altough we could relax Definition 2.1 to allow commutative *partial* monoids[5] such as the weight structure of probabilities $([0, 1], +, 0)$, not every weight function on $[0, 1]$ is a probability distribution. In fact, probabilistic systems (among others) can be recovered as ULTraSs over the $(\mathbb{R}_0^+, +, 0)$ (i.e. the free completion of $([0, 1], +, 0)$) and subject to suitable constraints. For instance, Segala systems [27] are precisely the strict subclass of $\mathbb{R}_0^+$-ULTraS such that every weight function ρ in their transition relation is a probability distribution i.e. $\llbracket\rho\rrbracket = 1$. Moreover, bisimulation is preserved by constraints; e.g., bisimulations on the above class of (constrained) ULTraS corresponds to Segala's (strong) bisimulations [27, Def. 13].

Proposition 2.8. *Let $(X, A, \rightarrowtriangle_X)$ and $(Y, A, \rightarrowtriangle_Y)$ be image-finite Segala-systems seen as ULTraSs on $(\mathbb{R}_0^+, +, 0)$. Every bisimulation relation between them is a strong bisimulation in the sense of [27, Def. 13] and vice versa.*

Proof (Omitted). See Appendix B. □

A similar result holds for generative (or fully) or reactive probabilistic systems and their bisimulations. In fact, these are functional $\mathbb{R}_+^0$-ULTraS s.t. for all $x \in X$ $x \xrightarrow{a}\!\!\!\triangleright \rho \implies \llbracket\rho\rrbracket \in \{0, 1\}$ and $\sum_{\{\rho \mid x \xrightarrow{a}\triangleright \rho\}} \llbracket\rho\rrbracket \in \{0, 1\}$ respectively.

[5] A commutative partial monoid is a set endowed with a unit and a partial binary operation which is associative and commutative, where it is defined, and always defined on its unit.

2.3 Comparison with $\mathcal{M}$-bisimulation

Bernardo et al. defined a notion of bisimulation for ULTraS parametrized by a function $\mathcal{M}$ which is used to weight sets of (sequences of) transitions [5, Def. 3.3]. Notably, $\mathcal{M}$'s codomain may be not the same of that used for weight functions in the transition relation. This offers an extra degree of freedom with respect to Definition 2.6. We recall the relevant definitions with minor modifications since the original ones have to consistently weight also sequences of transitions in order to account also for trace equivalences which are not in the scope of this paper.

Definition 2.9 (M-function)**.** *Let* $(M, \perp)$ *be a pointed*[6] *set and* $(X, A, \rightarrowtriangle)$ *be a* $\mathfrak{W}$*-ULTraS. A function* $\mathcal{M} : X \times A \times \mathcal{P}X \to M$ *is an* M-function for $(X, A, \rightarrowtriangle)$ *if, and only if, it agrees with termination and class union, i.e.:*

- *for all* $x \in X$*,* $a \in A$ *and* $C \in \mathcal{P}X$*,* $\mathcal{M}(x, a, C) = \perp$ *whenever* $x \not\xrightarrow{a}\!\!\rhd$ *or* $\llbracket \rho\big|_C \rrbracket = 0$ *for every* $x \xrightarrow{a}\!\!\rhd \rho$*;*
- *for all* $x, y \in X$*,* $a \in A$ *and* $C_1, C_2 \in \mathcal{P}X$*, if* $\mathcal{M}(x, a, C_1) = \mathcal{M}(y, a, C_1)$ *and* $\mathcal{M}(x, a, C_2) = \mathcal{M}(y, a, C_2)$ *then* $\mathcal{M}(x, a, C_1 \cup C_2) = \mathcal{M}(y, a, C_1 \cup C_2)$*.*

Definition 2.10 ($\mathcal{M}$-bisimulation [5])**.** *Let* $\mathcal{M}$ *be an* M*-function for* $(X, A, \rightarrowtriangle)$*. An equivalence relation* $R \subseteq X \times X$ *is a* $\mathcal{M}$-bisimulation for $\rightarrowtriangle$ *iff for each pair* $(x, y) \in R$*, label* $a \in A$*, and class* $C \in X/R$*,* $\mathcal{M}(x, a, C) = \mathcal{M}(y, a, C)$*.*

Differently from Definition 2.6, M may be not W allowing one to, for instance, consider stochastic rates up-to a suitable tolerance as a way to account for experimental measurement errors in the model. A further distinction between bisimulation and $\mathcal{M}$-bisimulation arises from the fact that ULTraSs come with two distinct ways of *terminating*. A state can be seen as “terminated” either when its outgoing transitions are always the constantly zero function, or when it has no transitions at all. In the first case, the state has still associated an outcome, saying that no further state is reachable; we call these states *terminal*. In the second case, the LTS does not even tell us that the state cannot reach any further state; in fact, there is no “meaning” associated to the state. In this case, we say that the state is *stuck*.[7] The bisimulation given in Definition 2.6 keeps these two terminations as different (i.e., they are not bisimilar), whereas $\mathcal{M}$-bisimulation does not make this distinction (cf. [5, Def. 3.2] or, for a concrete example based on Segala systems, [5, Def. 7.2]).

Finally, the two notions differ on the quantification over equivalence classes: in the case of Definition 2.6 quantification depends on the non-deterministic step whereas in the case of $\mathcal{M}$-bisimulation it does not.

Under some mild assumptions, the two notions agree. In particular, let us restrict to systems with just one of the two terminations for each action a—i.e. if for some x, $\{\rho \mid x \xrightarrow{a}\!\!\rhd \rho\} = \emptyset$ then for all y, $\lambda z.0 \notin \{\rho \mid y \xrightarrow{a}\!\!\rhd \rho\}$, and, symmetrically, if for some x, $\lambda z.0 \in \{\rho \mid x \xrightarrow{a}\!\!\rhd \rho\}$ then for all y, $\{\rho \mid y \xrightarrow{a}\!\!\rhd \rho\} \neq \emptyset$. Then, the bisimulation given in Definition 2.6 corresponds to a $\mathcal{M}$-bisimulation for a suitable choice of $\mathcal{M}$.

Proposition 2.11. *Let* $(X, A, \rightarrowtriangle)$ *be a* $\mathfrak{W}$*-ULTraS with at most one kind of termination, for each label. Every bisimulation* R *is also an* $\mathcal{M}$*-bisimulation for*

$$\mathcal{M}(x, a, C) \triangleq \{[\rho]_{R_{\mathfrak{W}}} \mid x \xrightarrow{a}\!\!\rhd \rho \text{ and } \llbracket \rho\big|_C \rrbracket \neq 0\} \cup \{[\lambda z.0]_{R_{\mathfrak{W}}}\}$$

[6] A pointed set (sometimes called based set or rooted set) is a set equipped with a distinguished element called (base) point; homomorphisms are point preserving functions.

[7] This is akin to sequential programs: a terminal state is when we reach the end of the program; a stuck state is when we are executing an instruction whose meaning is undefined.

where $(M, \bot) = (\mathcal{P}_f(\mathcal{F}_{\mathfrak{W}} X/R_{\mathfrak{W}}), \{[\lambda z.0]_{R_{\mathfrak{W}}}\})$.

Proof (Omitted). See Appendix C. □

Intuitively, Definition 2.6 generalises strong bisimulation for Segala systems (Segala and Lynch's probabilistic bisimilarity [27]) and $\mathcal{M}$-bisimulation generalises convex bisimulation [5].

3 WF-GSOS: A complete GSOS format for ULTraSs

In this section we introduce the *Weight Function SOS* specification format for the syntactic presentation of ULTraSs. As it will be proven in Section 5.3, bisimilarity for systems given in this format is guaranteed to be a congruence with respect to the signature used for representing processes.

The format is parametric in the weight monoid $\mathfrak{W}$ and, as usual, in the *process signature* Σ defining the syntax of system processes. In contrast with "classic" GSOS formats [19], targets of rules are not processes but terms whose syntax is given by a different signature, called the *weight signature*. This syntax can be thought of as an "intermediate language" for representing weight functions along the line of viewing ULTraSs as stratified (or staged) systems. An early example of this approach can be found in [2], where targets are terms representing measures over the continuous state space. Earlier steps in this direction can be found e.g. in Bartels' GSOS format for Segala systems (cf. [4, §5.3] and [24, §4.2]) or in [10, 5] where targets are described by meta-expressions.

Definition 3.1 (WF-GSOS Rule)**.** *Let* $\mathfrak{W}$ *be a commutative monoid and* A *a set of labels. Let* Σ *and* Θ *be the* process signature *and the* weight signature*, respectively. A WF-GSOS rule over them is a rule of the form:*

$$\frac{\left\{x_i \xrightarrow{a}\!\!\!\triangleright\ \phi^a_{ij}\right\}^{1\le i\le n,}_{\substack{a\in A_i,\\ 1\le j\le m^a_i}} \quad \left\{x_i \not\xrightarrow{b}\!\!\!\triangleright\ \right\}_{\substack{1\le i\le n,\\ b\in B_i}} \quad \left\{\llbracket \phi^{a_k}_{i_k j_k}\rrbracket = w_k\right\}_{1\le k\le p} \quad \left\{\phi^{a_k}_{i_k j_k}\Big|^{\mathfrak{C}_k} \ni y_k\right\}_{1\le k\le q}}{\mathtt{f}(x_1,\dots,x_n) \xrightarrow{c}\!\!\!\triangleright\ \psi}$$

where:

- $\mathtt{f}$ *is an* n*-ary symbol from* Σ*;*
- $X = \{x_i \mid 1 \le i \le n\}$, $Y = \{y_k \mid 1 \le k \le q\}$ *are sets of pairwise distinct* process *variables;*
- $\Phi = \{\phi^a_{ij} \mid 1 \le i \le n,\ a \in A_i,\ 1 \le j \le m^a_i\}$ *is a set of pairwise distinct* weight function *variables;*
- $\{w_k \in \mathfrak{W} \mid 1 \le k \le p\}$ *are* weight constants*;*
- $\{\mathfrak{C}_k \mid 1 \le k \le q,\ w_k \in \mathfrak{C}_k\}$ *is a set of* clubs *of* $\mathfrak{W}$*, i.e. subsets of* W *being monoid ideals whose complements are sub-monoids of* $\mathfrak{W}$*;*
- $a, b, c \in A$ *are labels and* $A_i \cap B_i = \emptyset$ *for* $1 \le i \le n$*;*
- ψ *is a* weight term *for the signature* Θ *such that* $\mathrm{var}(\psi) \subseteq X \cup Y \cup \Phi$.

A rule like above is triggered *by a tuple* $\langle C_1, \dots, C_n\rangle$ *of* enabled labels *and by a tuple* $\langle v_1, \dots, v_p\rangle$ *of weights if, and only if,* $A_i \subseteq C_i$*,* $B_i \cap C_i = \emptyset$*, and* $w_j = v_j$ *for* $1 \le i \le n$ *and* $1 \le j \le p$.

Intuitively, the four families of premises can be grouped in two kinds: the first two families correspond to the non-deterministic (and labelled) behaviour, whereas the other two correspond to the weighting behaviour of quantitative aspects. The former are precisely the premises of

GSOS rules for LTSs (up-to targets being functions), and describe the possibility to perform some labelled transitions. The latter are inspired by Bartels' *Segala-GSOS* [4, §5.3] and Klin's WGSOS [21] formats; a premise like $\llbracket\phi\rrbracket = w$ constrains the variable ϕ to those functions whose total weight is exactly the constant w; a premise like $\phi\big|^{\mathfrak{C}} \ni y$ binds the process variable y to those elements being assigned a weight in $\mathfrak{C}$. This kind of premises are meant to single out elements from weight functions domain in a way that is coherent w.r.t. function actions (hence independent from carrier maps and variable substitutions). To this end, selection may depend on weights only and has to be unaffected by sums, i.e., $z = f(x) = f(x')$ is selected if and only if at least x or x' is. Clubs are the finest substructures of commutative monoids that are "isolated" w.r.t. the monoidal operation in the sense that:

- are commutative monoid ideals, i.e. subsets $\mathfrak{C}$ with a module structure;
- their complement $\overline{\mathfrak{C}}$ in $\mathfrak{W}$ is a sub-monoid of $\mathfrak{W}$.

Because of the first assumption $v + w \in \mathfrak{C} \implies v \in \mathfrak{C} \vee w \in \mathfrak{C}$ and because of the second $v, w \notin \mathfrak{C} \implies v + w \notin \mathfrak{C}$ In other words, if something is selected depending on its weight, no matter what is added to, it will remain selected and vice versa: $v + w \in \mathfrak{C} \iff v \in \mathfrak{C} \vee w \in \mathfrak{C}$. Note that no club can contain the unit 0 (otherwise $\overline{\mathfrak{C}} = \emptyset$) and this ensures selections to be confined within the weight function supports (hence to be finite).

Remark 3.2. The empty set trivially is a club. Not all complements of submonoids are clubs, for instance even natural numbers under addition are a submonoid of $(\mathbb{N}, +, 0)$ but odd numbers are not a club; the only non-empty club in $(\mathbb{N}, +, 0)$ is $\mathbb{N} \setminus \{0\}$. Elements with an opposite cannot be part of a club: if $x \in \mathfrak{C}$ then $x + (-x) = 0$ is in $\mathfrak{C}$ and hence $\overline{\mathfrak{C}}$ cannot be a submonoid of $\mathfrak{W}$.

Like Segala-GSOS (but unlike WGSOS), there are no variables denoting the weight of each y_k since this information can be readily extracted from $\phi^{a_k}_{i_k j_k}$, e.g. by some operator from Θ that "evaluates" $\phi^{a_k}_{i_k j_k}$ on y_k. Targets of transitions defined by these rules are terms generated from the signature Θ. In order to characterize transition relations for ULTraSs, we need to *evaluate* these terms to weight functions. This is obtained by adding an *interpretation for weight terms*, besides a set of rules in the above format.

Before defining interpretations and specifications, we need to introduce some notation. For a signature S and a set X of variable symbols, let $T^S X$ denote the set of terms freely generated by S over the variables X (in the following, S will be either Σ or Θ). A substitution for symbols in X is any function $\sigma : X \to Y$; its action extends to terms defining the function $T^S(\sigma) : T^S X \to T^S Y$ (i.e. simultaneous substitution). When confusion seems unlikely we use the more evocative notation $\mathtt{t}[\sigma]$ instead of $T^S(\sigma)(\mathtt{t})$.

Definition 3.3 (Interpretation)**.** *Let $\mathfrak{W}$ be a commutative monoid, Σ and Θ be the process and the weight signature respectively. A* weight term interpretation *for them is a family of functions*

$$\{\!\!|\text{-}|\!\!\}_X : T^{\Theta}(X + \mathcal{F}_{\mathfrak{W}}(X)) \to \mathcal{P}_f \mathcal{F}_{\mathfrak{W}} T^{\Sigma}(X)$$

indexed over sets of variable symbols, and respecting substitutions, i.e.:

$$\forall \sigma : X \to Y, \psi \in T^{\Theta}(X) : \{\!\!|\psi|\!\!\}_X[\sigma] = \{\!\!|\psi[\sigma]|\!\!\}_Y.$$

Different from [24] interpretations allow one term to represent finitely many weight functions. This generalization offers more freedom in the use of the format by reducing the constrains on what can be encoded in weight function terms and simplifies the proof for completeness.

We are ready to introduce the WF-GSOS specification format. Basically, this is a set of WF-GSOS rules, subject to some finiteness conditions to ensure image-finiteness, together with an interpretation.

Definition 3.4 (WF-GSOS specification)**.** *Let $\mathfrak{W}$ be a commutative monoid, A the set of labels, Σ and Θ the process and the weight signature respectively. An* image-finite WF-GSOS specification over $\mathfrak{W}, A, \Sigma$ and Θ *is a pair $\langle \mathcal{R}, \{\!|\text{-}|\!\}\rangle$ where $\{\!|\text{-}|\!\}$ is a weight term interpretation and $\mathcal{R}$ is a set of rules compliant with Definition 3.1 and such that only finitely many rules share the same operator in the source ($\mathtt{f}$), the same label in the conclusion (c), and the same trigger $\langle A_1, \dots, A_n\rangle$, $\langle w_1, \dots, w_p\rangle$.*

Every WF-GSOS specification induces an ULTraS over ground process terms.

Definition 3.5 (Induced ULTraS)**.** *The ULTraS induced by an image-finite WF-GSOS specification $\langle \mathcal{R}, \{\!|\text{-}|\!\}\rangle$ over $\mathfrak{W}, \Sigma, \Theta$ is the $\mathfrak{W}$-ULTraS $(T^{\Sigma}\emptyset, A, \rightarrowtriangle)$ where $\rightarrowtriangle$ is defined as the smallest subset of $T^{\Sigma}\emptyset \times A \times \mathcal{F}_{\mathfrak{W}}T^{\Sigma}\emptyset$ being closed under the following condition.*

Let $p = \mathtt{f}(p_1, \dots, p_n) \in T^{\Sigma}\emptyset$. Since the ground Σ-terms p_i are structurally smaller than p assume (by structural recursion) that the set $\{\rho \mid p_i \xrightarrow{a}\!\!\triangleright \rho\}$ – and hence the trigger $\vec{A} = \langle A_1, \dots, A_n\rangle$, $\vec{w} = \langle w_1, \dots, w_q\rangle$ – is determined for every $i \in \{1, \dots, n\}$ and $a \in A$. For any rule $R \in \mathcal{R}$ whose conclusion is of the form $\mathtt{f}(x_1, \dots, x_n) \xrightarrow{c}\!\!\triangleright \psi$ and triggered by $\vec{A}$ and $\vec{w}$ let X, Y, Φ be the set of process and weight function variables involved in R as per Definition 3.1. Then, for any substitution $\sigma : X \cup Y \to T^{\Sigma}\emptyset$ and map $\theta : \Phi \to \mathcal{F}_{\mathfrak{W}}T^{\Sigma}\emptyset$ such that:

1. *$\sigma(x_i) = p_i$ for $x_i \in X$;*
2. *$\theta(\phi^a_{ij}) = \rho$ for each premise $x_i \xrightarrow{a}\!\!\triangleright \phi^a_{ij}$ and $\llbracket \phi^a_{ij} \rrbracket = w_k$ of R, and for any ρ such that $p_i \xrightarrow{a}\!\!\triangleright \rho$ and $\llbracket \rho \rrbracket = w_k$;*
3. *$\sigma(y_k) = q_k$ for each premise $\phi^{a_k}_{i_k j_k}\big|^{\mathfrak{C}_k} \ni y_k$ of R and for any $q_k \in T^{\Sigma}\emptyset$ s.t. $\theta(\phi^{a_k}_{i_k j_k})(q_k) \in \mathfrak{C}_k$;*

there is $p \xrightarrow{c}\!\!\triangleright \rho$ where $\rho \in \{\!|\psi[\theta]|\!\}_{X \cup Y}[\sigma]$ is an instantiated interpretation of the target Θ-term ψ.

The above definition is well-defined since it is based on structural recursion over ground Σ-terms (i.e. the process p in each triple (p, a, ρ)); in particular, terms have finite depth and only structurally smaller terms are used by the recursion (i.e. the assumption of $p_i \xrightarrow{a}\!\!\triangleright \rho$ being defined for each p_i in $p = \mathtt{f}(p_1, \dots, p_n)$). Moreover, for any trigger, operator, and conclusion label only finitely many rules have to be considered.

Finally we can state the main result for the proposed format.

Theorem 3.6 (Congruence)**.** *The bisimulation on the ULTraS induced by a WF-GSOS specification is a congruence with respect to the process signature.*

The proof is postponed to Section 5.3, where we will take advantage of the bialgebraic framework.

Remark 3.7 (Expressing interpretations). Weight term interpretation can be defined in many ways, e.g. by structural recursion on Θ-terms. For instance, every substitution-respecting family of maps:

$$h_X : \Theta\mathcal{F}_{\mathfrak{W}}T^{\Sigma}(X) \to \mathcal{P}_f\mathcal{F}_{\mathfrak{W}}T^{\Sigma}(X) \qquad b_X : X \to \mathcal{P}_f\mathcal{F}_{\mathfrak{W}}T^{\Sigma}(X)$$

uniquely extends to an interpretation by structural recursion on Θ-terms where h_X and b_X define the inductive and base cases respectively. These maps can be easily given by means of a set of equations, as in [24, §4.1].

$$\frac{}{(a,r).P \xrightarrow{a,r} P} \quad \frac{P_1 \xrightarrow{a,r} Q}{P_1{+}P_2 \xrightarrow{a,r} Q} \quad \frac{P_2 \xrightarrow{a,r} Q}{P_1{+}P_2 \xrightarrow{a,r} Q} \quad \frac{P \xrightarrow{a,r} Q}{P\backslash L \xrightarrow{a,r} Q}\ a \notin L \quad \frac{P \xrightarrow{a,r} Q}{P\backslash L \xrightarrow{\tau,r} Q}\ a \in L$$

$$\frac{P_1 \xrightarrow{a,r_1} Q_1 \quad P_2 \xrightarrow{a,r_2} Q_2}{P_1 \underset{L}{\bowtie} P_2 \xrightarrow{a,R} Q_1 \underset{L}{\bowtie} Q_2}\ a \in L \quad \frac{P_1 \xrightarrow{a,r} Q}{P_1 \underset{L}{\bowtie} P_2 \xrightarrow{a,r} Q \underset{L}{\bowtie} P_2}\ a \notin L \quad \frac{P_2 \xrightarrow{a,r} Q}{P_1 \underset{L}{\bowtie} P_2 \xrightarrow{a,r} P_1 \underset{L}{\bowtie} Q}\ a \notin L$$

Figure 1: Structural operational semantics for PEPA.

4 Examples and applications of WF-GSOS specifications

In this section we provide some examples of applications of the WF-GSOS format. First, we show how a process calculus can be given a WF-GSOS specification; in particular, we consider PEPA, a well known process algebra with quantitative features. Then we show that Klin's Weighted GSOS format for weighted systems [21] and Bartels' Segala-GSOS format for Segala systems [4] are subsumed by our WF-GSOS format; this corresponds to the fact that ULTraSs subsume both weighted and Segala systems.

4.1 WF-GSOS for PEPA

In PEPA [16, 17], processes are terms over the grammar:

$$P ::= (a,r).P \mid P + P \mid P \underset{L}{\bowtie} P \mid P \setminus L \tag{2}$$

where a ranges over a fixed set of labels A, L over subsets of A and r over $\mathbb{R}^+$. The semantics of process terms is usually defined by the inference rules in Figure 1 where $a \in A$, $r, r_1, r_2, R \in \mathbb{R}^+$ (passive rates are omitted for simplicity) and R depends only on r_1, r_2 and the intended meaning of synchronisation. For instance, in applications to performance evaluation [16], rates model time and R is defined by the *minimal rate law*:

$$R = \frac{r_1}{r_a(P_1)} \cdot \frac{r_2}{r_a(P_2)} \cdot \min(r_a(P_1), r_a(P_2)) \tag{3}$$

where r_a denotes the apparent rate of a [16].

PEPA can be characterized by a specification in the WF-GSOS format where the process signature Σ is the same as (2) and weights are drawn from the monoid of positive real numbers under addition extended with the $+\infty$ element (only for technical reasons connected with the $\{\!|\text{-}|\!\}$ and process variables—differently from other stochastic process algebras like EMPA [6], PEPA does not allow instantaneous actions, i.e. with rate $+\infty$). The intermediate language of weight terms is expressed by the grammar:

$$\theta ::= \perp \mid \Diamond_r(\theta) \mid \theta_1 \oplus \theta_2 \mid \theta_1 \parallel_L \theta_2 \mid \xi \mid P$$

where $r \in \mathbb{R}_0^+$, $L \subseteq A$, ξ range over weight functions $\mathcal{F}_{\mathfrak{W}}X$, and P over processes in $T^\Sigma X$ for some set X. Note that the grammar is untyped since it describes the terms freely generated by the signature $\Theta = \{\perp: 0, \Diamond_r : 1, \oplus : 2, \parallel_L: 2\}$, over weight function variables and processes. Intuitively $\perp$ is the constantly 0 function, $\Diamond_r$ reshapes its argument to have total weight r, $\oplus$ is the point-wise sum and $\parallel_L$ parallel composition e.g. by (3). The formal meaning of these operators is given below by the definition (by structural recursion on Θ-terms) of the interpretation $\{\!|\text{-}|\!\}$ which is introduced alongside WF-GSOS rules for presentation convenience. Each operator is interpreted as a singleton (PEPA describes functional ULTraSs) and hence we will describe $\{\!\!\{-\}\!\!\}$ as if a weight function is returned.

For each action $a \in A$ and rate $r \in \mathbb{R}^+$, a process $(a,r).P$ presents exactly one a-labelled transition ending in the weight function assigning r to the (sub)process denoted by the variable P and 0 to everything else. Hence, the *action axiom* is expressed as follows:

$$\frac{}{(a,r).P \xrightarrow{a}\!\!\triangleright\ \Diamond_r(P)} \qquad \{\!\!\{\Diamond_r(\psi)\}\!\!\}_X(t) = \begin{cases} \frac{r}{\lceil\lfloor \{\!\!\{\psi\}\!\!\}_X \rfloor\rceil} & \text{if } \{\!\!\{\psi\}\!\!\}_X(t) \neq 0 \\ 0 & \text{otherwise} \end{cases}$$

where $\Diamond_r$ normalises[8] $\{\!\!\{P\}\!\!\}_X$ to equally distribute the weight r over its support; in particular, since process variables will be interpreted as "Dirac-like" functions $\Diamond_r(P)$ corresponds to the weight function assigning r to Σ-term denoted by P.

Conversely to the action axiom, $(a,r).P$ can not perform any action but a:

$$\frac{}{(a,r).P \xrightarrow{b}\!\!\triangleright\ \perp}\, a \neq b \qquad \{\!\!\{\perp\}\!\!\}_X(t) = 0$$

This rule is required to obtain a functional ULTraS and is implicit in Figure 1 where disabled transitions are assumed with rate 0 as in any specification in the Stochastic GSOS or Weighted GSOS formats. Without this rule, transitions would have been disabled in the non-deterministic layer i.e. $(a,r).P \not\xrightarrow{b}\!\!\triangleright$.

Stochastic choice is resolved by the stochastic race, hence the rate of each competing transition is added point-wise as in Figure 1 (and in the SGSOS and WGSOS formats). This passage belongs to the stochastic layer of the behaviour (hence to the interpretation, in our setting) whereas the selection of which weight functions to combine is in the non-deterministic behaviour represented by the rules and, in particular, to the labelling. Therefore, the *choice rules* become:

$$\frac{P_1 \xrightarrow{a}\!\!\triangleright\ \phi_1 \quad P_2 \xrightarrow{a}\!\!\triangleright\ \phi_2}{P_1 + P_2 \xrightarrow{a}\!\!\triangleright\ \phi_1 \oplus \phi_2} \qquad \{\!\!\{\psi \oplus \phi\}\!\!\}_X(t) = \{\!\!\{\psi\}\!\!\}_X(t) + \{\!\!\{\phi\}\!\!\}_X(t)$$

Likewise, process cooperation depends on the labels to select the weight function to be combined. This is done in the next two rules: one when the two processes cooperate, and the other when one process does not interact on the channel:

$$\frac{P_1 \xrightarrow{a}\!\!\triangleright\ \phi_1 \quad P_2 \xrightarrow{a}\!\!\triangleright\ \phi_2}{P_1 \underset{L}{\bowtie} P_2 \xrightarrow{a}\!\!\triangleright\ \phi_1 \parallel_L \phi_2}\, a \in L \qquad \frac{P_1 \xrightarrow{a}\!\!\triangleright\ \phi_1 \quad P_2 \xrightarrow{a}\!\!\triangleright\ \phi_2}{P_1 \underset{L}{\bowtie} P_2 \xrightarrow{a}\!\!\triangleright\ (\phi_1 \parallel_L P_2) \oplus (P_1 \parallel_L \phi_2)}\, a \notin L$$

The combination step depends on the minimal rate law (3):

$$\{\!\!\{\psi \parallel_L \phi\}\!\!\}_X(t) = \begin{cases} \frac{\{\!\!\{\psi\}\!\!\}_X(t_1)}{\llbracket \{\!\!\{\psi\}\!\!\}_X \rrbracket} \cdot \frac{\{\!\!\{\phi\}\!\!\}_X(t_2)}{\llbracket \{\!\!\{\phi\}\!\!\}_X \rrbracket} \cdot \min(\llbracket \{\!\!\{\psi\}\!\!\}_X \rrbracket, \llbracket \{\!\!\{\phi\}\!\!\}_X \rrbracket) & \text{if } t = t_1 \underset{L}{\bowtie} t_2 \\ 0 & \text{otherwise} \end{cases}$$

Each process is interpreted as a weight function over process terms. This is achieved by a Dirac-like function assigning $+\infty$ to the Σ-term composed by the aforementioned variable: $\{\!\!\{P\}\!\!\}_X(t) = +\infty$ if $P = t$, 0 otherwise. The infinite rate characterizes instantaneous actions as if

[8]Since the interpretation $\{\!\!\{-\}\!\!\}$ is being defined by structural recursion and has to cover all the language freely generated from Θ, we can not use the (slightly more intuitive) "Dirac" operator $\delta_r(P)$ where P is restricted to be a process variable instead of a Θ-term. Likewise, indexing $\delta_{r,P}$ also over processes would break substitution independence i.e. naturality.

all the mass is concentrated in the variable; e.g., in interactions based on the minimal rate law, processes are not consumed. For the same reason, if we were dealing with concentration rates and the multiplicative law, we would assign 1 to P.

The remaining rules for hiding are straightforward:

$$\frac{P \xrightarrow{a}\!\!\triangleright\ \phi}{P \setminus L \xrightarrow{a}\!\!\triangleright\ \phi}\ a \notin L \qquad \frac{P \xrightarrow{a}\!\!\triangleright\ \phi}{P \setminus L \xrightarrow{\tau}\!\!\triangleright\ \phi}\ a \in L$$

This completes the definition of $\{\!|\text{-}|\!\}$ by structural recursion and hence the WF-GSOS specification of PEPA. It is easy to check that the induced ULTraS is functional and correspond to the stochastic system of PEPA processes, that bisimulations on it are stochastic bisimulations (and vice versa) and that bisimilarity is a congruence with respect to the process signature.

4.2 Segala-GSOS

In [4], Bartels proposed a GSOS specification format[9] for Segala systems (hence Segala-GSOS), i.e. ULTraS where weight functions are exactly probability distributions. We recall Bartels' definition, with minor notational differences.

Definition 4.1 ([4, §5.3]). *A* GSOS rule for Segala systems *is a rule of the form*

$$\frac{\left\{x_i \xrightarrow{a} \phi^a_{ij}\right\}_{1\leq i\leq n,\ a\in A_i,\ 1\leq j\leq m^a_i} \quad \left\{x_i \not\xrightarrow{b}\right\}_{1\leq i\leq n,\ b\in B_i} \quad \left\{\phi^a_{ij} \Longrightarrow y_k\right\}_{1\leq k\leq q}}{\mathtt{f}(x_1,\ldots,x_n) \xrightarrow{c} w_1\cdot t_1 + \cdots + w_m \cdot t_m}$$

where:

- $\mathtt{f}$ *is an n-ary symbol from* Σ*;*
- $X = \{x_i \mid 1 \leq i \leq n\}$, $Y = \{y_k \mid 1 \leq k \leq q\}$, *and* $V = \{\phi^a_{ij} \mid 1 \leq i \leq n,\ a \in A_i,\ 1 \leq j \leq m^a_i\}$ *are pairwise distinct* process *and* probability distribution *variables respectively;*
- $a, b, c \in A$ *are labels and* $A_i \cap B_i = \emptyset$ *for any* $i \in \{1, \ldots, n\}$*;*
- $t_1, \ldots, t_m$ *are target terms on variables* X, Y *and* V*; the latter are associated with colours from a finite palette to indicate different instances;*
- $\{w_i \in (0, 1] \mid 1 \leq i \leq m\}$ *describe a linear composition of the targets terms i.e. are weights associated to the target terms and such that* $w_1 + \cdots + w_m = 1$*.*

A rule like above is triggered *by a tuple* $\langle C_1, \ldots, C_n\rangle$ *of* enabled labels *if, and only if,* $A_i \subseteq C_i$ *and* $B_i \cap C_i = \emptyset$ *for each* $i \in \{1, \ldots, n\}$*. A* GSOS specification for Segala systems *is a set of rules in the above format containing finitely many rules for any source symbol* $\mathtt{f}$*, conclusion label* c *and trigger* $\vec{C}$*.*

Segala-GSOS specifications can be easily turned into WF-GSOS ones. The first two families of premises are translated straightforwardly to the corresponding ones in our format; the third can be turned into those of the form $\lfloor\phi\rfloor \ni y$. Targets of transitions describe finite probability distributions and are evaluated to actual probability distributions by a fixed interpretation of a form similar to Definition 3.3. Some care is needed to handle copies of probability variables. In practice, duplicated variables are expressed by adding "colouring" operators to Θ; their number is finite and depends only on the set of rules since multiplicities are fixed and finite for rules in the above format. Let $\tilde{V}$ be the set of "coloured" variables from V where the colouring is

[9] Segala-GSOS specifications yield distributive laws for Segala systems but it still is an open problem whether every such distributive law arises from some Segala-GSOS specification.

used to distinguish duplicated variables (cf. [4, §5.3]). Given a substitution ν from $\tilde{V}$ to (finite) probability distributions over $T^{\Sigma}(X+Y)$, each t_i is interpreted as the probability distribution:

$$\tilde{t}_i(t) \triangleq \begin{cases} \prod_{k=1}^{|\tilde{V}\cap var(t_i)|} \nu(\phi_k)(t_k) & \text{if } t = t_i[\phi_k/t_k] \text{ for } t_k \in T^{\Sigma}(X+Y) \\ 0 & \text{otherwise} \end{cases}$$

and each target term $w_1 \cdot t_1 + \cdots + w_m \cdot t_m$ is interpreted as the convex combination of $\tilde{t}_1, \ldots, \tilde{t}_m$.

4.3 Weighted GSOS

In [21], Klin and Sassone proposed a GSOS format[10] for Weighted LTSs that is parametric in the commutative monoid $\mathfrak{W}$ and hence called $\mathfrak{W}$-GSOS. The format subsumes many known formats for systems expressible as $WLTS$: for instance, Stochastic GSOS specifications are in the $\mathbb{R}_0^+$-GSOS format and GSOS for LTS are in the $\mathbb{B}$-GSOS format where $\mathfrak{2} = (\{\mathtt{tt}, \mathtt{ff}\}, \vee, \mathtt{ff})$.

Definition 4.2 ([21, Def. 13]). *A $\mathfrak{W}$-GSOS rule is an expression of the form:*

$$\frac{\left\{x_i \xrightarrow{a}\!\!\lhd\; w_{ai}\right\}_{1\le i\le n,\ a\in A_i} \qquad \left\{x_{i_k} \xrightarrow{b_k,u_k}\!\!\rhd\; y_k\right\}_{1\le k\le m}}{\mathtt{f}(x_1,\ldots,x_n) \xrightarrow{c,\beta(u_1,\ldots,u_m)}\!\!\rhd\; t}$$

where:

- $\mathtt{f}$ *is an n-ary symbol from* Σ*;*
- $X = \{x_i \mid 1 \le i \le n\}$, $Y = \{y_k \mid 1 \le k \le m\}$ *and* $\{u_k \mid 1 \le k \le m\}$ *are pairwise distinct* process *and* weight *variables;*
- $\{w_{ai} \in \mathfrak{W} \mid 1 \le i \le n,\ a \in A_i\}$ *are* weight constants *such that* $w_{i_k} \neq 0$ *for* $1 \le k \le m$*;*
- $\beta : W^m \to W$ *is a multiadditive function on* $\mathfrak{W}$*;*
- $a, b, c \in A$ *are labels and* $A_i \subseteq A$ *for* $1 \le i \le n$*;*
- t *is a* Σ*-term such that* $Y \subseteq \mathrm{var}(\mathtt{t}) \subseteq X \cup Y$*;*

A rule is triggered *by a n-tuple* $\vec{C}$ *of* enabled labels *s.t.* $A_i \subseteq C_i$ *and by a family of weights* $\{v_{ai} \mid 1 \le i \le n,\ a \in A_i\}$ *s.t.* $w_{ai} = v_{ai}$*. A* $\mathfrak{W}$*-GSOS specification is a set of rules in the above format such that there are only finitely many rules for the same source symbol, conclusion label and trigger.*

Each rule describes the weight of $\mathtt{t}$ in terms of weights assigned to each y_k (i.e. u_k) occurring in it; if two rules share the same symbol, label, trigger and target then their contribute for $\mathtt{t}$ is added.

To turn a $\mathfrak{W}$-GSOS specification into WF-GSOS ones, the first step is to make weight function explicit, by means of premises like $x_i \xrightarrow{a}\!\!\rhd\; \phi_i^a$ (since WLTS are functional ULTraS, i.e. $m_i^a = 1$). Then, each premise $x_i \xrightarrow{a}\!\!\lhd\; w_{ai}$ is translated into $\llbracket \phi_i^a \rrbracket = w_{ai}$. If $\mathfrak{W}$ is *positive* (i.e., whenever $a+b=0$ then $a=b=0$) then $W \setminus \{0\}$ is a club and the translation of a $\mathfrak{W}$-GSOS into a WF-GSOS is straightforward. More generally, it suffices to combine rules sharing the same source, label and trigger into a single WF-GSOS rule with the same source, label and trigger. Its target is a suitable weight term containing the functions β and targets $\mathtt{t}$ of the original rules; every occurrence of variables y_k and u_k is replaced with the corresponding function variable (i.e. $\phi_{i_k}^{b_k}$). In order to deal with multiple copies of the same weight variable, we wrap each occurrence in a different "colouring" operator, like in the case of Segala-GSOS.

[10] Weighted GSOS specifications are proved to yield GSOS distributive laws for Weighted LTSs but it is currently an open question whether the format is also complete.

5 A coalgebraic presentation of ULTraS and WF-GSOS

The aim of this section is to prove some important results about WF-GSOS specifications. We first provide a characterization of ULTraSs as coalgebras for a specific behavioural functor (Section 5.2), and their bisimulations as *cocongruences*. Then, leveraging this characterization in Section 5.3 we apply Turi and Plotkin's bialgebraic theory [29], which allows us to define the categorical notion of *WF-GSOS distributive laws*; these laws describe the interplay between syntax and behaviour in any GSOS presentation of ULTraS. We will prove that every WF-GSOS specification yields a WF-GSOS distributive law, i.e., the format is *sound*. As a consequence, we obtain that the bisimilarities induced by these specifications are always congruence relations. Finally, in Section 5.4 we prove that WF-GSOS specification are also *complete*: every abstract WF-GSOS distributive law can be described by means of a WF-GSOS specification.

5.1 Abstract GSOS

In [29], Turi and Plotkin detailed an abstract presentation of well-behaved structural operational semantics for systems of various kinds. There syntax and behaviour of transition systems are modelled by algebras and coalgebras respectively. For instance, an (image-finite) LTS with labels in A and states in X is seen as a (successor) function $h : X \to (\mathcal{P}_f X)^A$ mapping each state x to a function yielding, for each label a, the (finite) set of states reachable from x via a-labelled transitions i.e. $\{y \mid x \xrightarrow{a} y\}$:

$$y \in h(x)(a) \iff x \xrightarrow{a} y.$$

Functions like h are *coalgebras* for the (finite) *labelled powerset functor* $(\mathcal{P}_f)^A$ over the category of sets and functions Set. In general, state based transition systems can be viewed as B-*coalgebra* i.e. sets (*carriers*) enriched by functions (*structures*) like $h : X \to BX$ for some suitable covariant functor $B : \mathsf{Set} \to \mathsf{Set}$. The Set-endofunctor B is often called *behavioural* since it encodes the computational behaviour characterizing the given kind of systems. A *morphism* from a B-coalgebra $h : X \to BX$ to $g : Y \to BY$ is a function $f : X \to Y$ such that the coalgebra structure h on X is consistently mapped to the coalgebra structure g on Y i.e. $g \circ f = Bf \circ h$. Therefore, B-coalgebras and their homomorphisms form the category B-CoAlg.

Two states $x, y \in X$ are said to be *behaviourally equivalent* with respect to the coalgebraic structure $h : X \to BX$ if they are equated by some coalgebraic morphism from h. Behavioural equivalences are generalised to two (or more) systems in the form of kernel bisimulations [28] i.e. as the pullbacks of morphisms extending to a cospan for the B-coalgebas structures associated with the given systems as pictured below.

$$\begin{tikzcd}
 & R \arrow[dl, "p_1"'] \arrow[dr, "p_2"] & \\
X_1 \arrow[dr, "f_1"] \arrow[dd, "h_1"'] & & X_2 \arrow[dl, "f_2"'] \arrow[dd, "h_2"] \\
 & Y \arrow[dd, "g"'] & \\
BX_1 \arrow[dr, "Bf_1"'] & & BX_2 \arrow[dl, "Bf_2"] \\
 & BY &
\end{tikzcd}$$

If the cospan f_1, f_2 is jointly epic, i.e. $j \circ f_1 = k \circ f_2 \implies j = k$ for any $j, k : C \to Z$, (in general if $\{f_i\}$ is an epic sink, hence $\{p_i\}$ is a monic source) then the set Y is isomorphic to the equivalence classes induced by R. We refer the interested reader to [26] for more information on the coalgebraic approach to process theory.

Dually, process syntax is modelled via algebras for endofunctors. Every algebraic signature Σ defines an endofunctor $\Sigma X = \coprod_{\mathtt{f} \in \Sigma} X^{ar(\mathtt{f})}$ on Set such that every model for the signature is an algebra for the functor i.e. a set X (carrier) together with a function $g : \Sigma X \to X$ (structure). A morphism from a Σ-algebras $g : \Sigma X \to X$ to $h : \Sigma Y \to Y$ is a function $f : X \to Y$ such that $f \circ g = h \circ \Sigma f$. The set of Σ-terms with variables from a set X is denoted by $T^{\Sigma}X$ and the set of ground ones admits an obvious Σ-algebra $a : \Sigma T^{\Sigma}\emptyset \to T^{\Sigma}\emptyset$ which is the *initial* Σ*-algebra* in the sense that for every other Σ-algebra g, there exists a unique morphism from a to g i.e. the *inductive extension* of the underlying function $f : T^{\Sigma}\emptyset \to X$. The construction T^{Σ} is a functor, moreover, it is the monad freely generated by Σ.

In [29], Turi and Plotkin showed that structural operational specifications for LTSs in the well-known image finite GSOS format [7] correspond to natural transformations of the following form:

$$\lambda : \Sigma(\mathrm{Id} \times B) \Longrightarrow BT^{\Sigma}.$$

These transformations, hence called *GSOS distributive laws*, contain the information needed to connect Σ-algebra and B-coalgebra structures over the same carrier set and capture the interplay between syntax and dynamics at the core of the SOS approach. These structures are called λ*-bialgebras* and are formed by a carrier X endowed with a Σ-algebra g and a B-coalgebra h structure s.t.:

$$\begin{tikzcd} \Sigma X \arrow[r, "g"] \arrow[d, "\Sigma\langle id_X, h\rangle"'] & X \arrow[r, "h"] & BX \\ \Sigma(X \times BX) \arrow[rr, "\lambda_X"] & & BT^{\Sigma}X \arrow[u, "Bg^{\flat}"'] \end{tikzcd}$$

where $g^{\flat} : T^{\Sigma}X \to X$ is the canonical extension of g by structural recursion. In particular, every λ-distributive law gives rise to a B-coalgebra structure over the set of ground Σ-terms $T^{\Sigma}\emptyset$ and to a Σ-algebra structure on the carrier of the final B-coalgebra. These two structures are part of the initial and final λ-bialgebra respectively and therefore, because the unique morphism from the former to the latter is both a Σ-algebra and a B-coalgebra morphism, observational equivalence on the system induced over $T^{\Sigma}\emptyset$ is a congruence with respect to the syntax Σ.

5.2 ULTraSs as coalgebras

Since ULTraSs alternate non-deterministic steps with quantitative steps, the corresponding behavioural functor can be obtained by composing the usual functor $(\mathcal{P}_f)^A : \mathsf{Set} \to \mathsf{Set}$ of non-deterministic labelled transition systems with the functors capturing the quantitative computational aspects: $\mathcal{F}_{\mathfrak{W}}$. It is easy to see that the action of a set function on a weight function (1) preserves identities and composition rendering $\mathcal{F}_{\mathfrak{W}}$ an endofunctor over Set.

For any $\mathfrak{W}$ and any A, A-labelled image-finite $\mathfrak{W}$-ULTraSs and their homomorphisms clearly form a category: $\mathsf{ULTS}_{\mathfrak{W},A}$. Objects and morphisms of this category are in 1-1 correspondence with $(\mathcal{P}_f\mathcal{F}_{\mathfrak{W}})^A$-coalgebras and their homomorphisms respectively.

Proposition 5.1. $\mathsf{ULTS}_{\mathfrak{W},A} \cong (\mathcal{P}_f\mathcal{F}_{\mathfrak{W}})^A\text{-}\mathsf{CoAlg}$.

Proof. Any image-finite $\mathfrak{W}$-ULTraS $(X, A, \rightarrowtriangle)$ determines a coalgebra (X, h) where, for any $x \in X$ and $a \in A$: $h(x)(a) \triangleq \{\rho \mid x \xrightarrow{a}\!\!\triangleright\, \rho\}$. Image-finiteness guarantees that these sets are finite and that their elements are finitely supported weight functions from X to the carrier of $\mathfrak{W}$. Then, it is easy to check that the correspondence is bijective. ◻

A similar result holds for the bisimulation given in Definition 2.6. Categorically, a relation between X and Y is a (jointly monic) span $X \leftarrow R \rightarrow Y$. In our case, this span has to be subject to some conditions, as shown next.

Proposition 5.2. *Let $(X_1, A, \rightharpoonup_1)$ and $(X_2, A, \rightharpoonup_2)$ be two image-finite $\mathfrak{W}$-ULTraSs; let (X_1, h_1), (X_2, h_2) be the corresponding coalgebras according Proposition 5.1. A relation between X_1 and X_2 is a bisimulation iff there exists a coalgebra (Y, g) and two coalgebra morphisms $f_1 : (X_1, h_1) \to (Y, g)$ and $f_2 : (X_2, h_2) \to (Y, g)$ such that f_1, f_2 are jointly epic and R is their pullback, i.e. the diagram below commutes.*

$$
\begin{tikzcd}
& R \arrow[dl, "p_1"'] \arrow[dr, "p_2"] \arrow[dd, phantom, "\lrcorner"] & \\
X_1 \arrow[dr, "f_1"] \arrow[dd, "h_1"'] & & X_2 \arrow[dl, "f_2"'] \arrow[dd, "h_2"] \\
& Y \arrow[dd, "g"] & \\
(\mathcal{P}_f \mathcal{F}_{\mathfrak{W}} X_1)^A \arrow[dr, "(\mathcal{P}_f \mathcal{F}_{\mathfrak{W}} f_1)^A"'] & & (\mathcal{P}_f \mathcal{F}_{\mathfrak{W}} X_2)^A \arrow[dl, "(\mathcal{P}_f \mathcal{F}_{\mathfrak{W}} f_2)^A"] \\
& (\mathcal{P}_f \mathcal{F}_{\mathfrak{W}} Y)^A &
\end{tikzcd}
$$

Proof (Omitted). See Appendix C. □

Intuitively, the system (Y, g) "subsumes" both (X_1, h_1) and (X_2, h_2) via f_1, f_2; then, R relates the states which are mapped to the same behaviour in Y ($(x_1, x_2) \in R$ iff $f_1(x_1) = f_2(x_2)$).

Coalgebraic bisimulation In Concurrency Theory also Aczel-Medler's *coalgebraic bisimulation* [1] is widely used. In fact, it is known that kernel bisimulations and coalgebraic bisimulations coincide if the behavioural functor is *weak pullback preserving* (wpp). This is the case for many behavioural functors, but not for $\mathcal{F}_{\mathfrak{W}}$ in general [21]. Actually, the fact that $\mathcal{F}_{\mathfrak{W}}$ (and $(\mathcal{P}_f \mathcal{F}_{\mathfrak{W}})^A$) preserves weak pullbacks depends on the underlying monoid only.

Definition 5.3. *A commutative monoid is called* positive *(sometimes zerosumfree, positively ordered) whenever $x + y = 0 \implies x = y = 0$ holds true. It is called* refinement *if for each $r_1 + r_2 = c_1 + c_2$ there is a 2×2 matrix $(m_{i,j})$ s.t. $r_i = m_{i,1} + m_{i,2}$ and $c_j = m_{1,j} + m_{2,j}$.*

Lemma 5.4. *Coalgebraic bisimulation and behavioural equivalence on ULTraSs coincides if $\mathfrak{W}$ is a positive refinement monoid.*

Proof. $(\mathcal{P}_f)^A$ is wpp, and under the lemma hypothesis also $\mathcal{F}_{\mathfrak{W}}$ is wpp, by [15]. Therefore, $(\mathcal{P}_f \mathcal{F}_{\mathfrak{W}})^A$ is wpp, hence every behavioural equivalence is a coalgebraic bisimulation on $(\mathcal{P}_f \mathcal{F}_{\mathfrak{W}})^A$-coalgebras. We conclude by Proposition 5.1. □

This condition, can be easily verified and in fact holds for several monoids of interest, e.g.: $(\{\mathtt{tt}, \mathtt{ff}\}, \vee, \mathtt{tt})$, $(\mathbb{N}, +, 0)$, $(\mathbb{R}_0^+, +, 0)$, $(\mathbb{N}, \max, 0)$, and $(A^*, \cdot, \varepsilon)$. A simple counter example is $(\{0, a, b, 1\}, +, 0)$ where $x + y \triangleq 1$ whenever $x \neq 0 \neq y$ for it is positive but not refinement (cf. $\vec{r} = \langle a, a\rangle$ and $\vec{c} = \langle b, b\rangle$).

5.3 WF-GSOS specifications are WF-GSOS distributive laws

In this subsection we put the WF-GSOS format within the bialgebraic framework [29]. As a consequence, we obtain that the bisimilarity induced by the ULTraS defined by this specification is a congruence.

In particular, we prove that every WF-GSOS specification represents a distributive law of the signature over the ULTRaS behavioural functor, i.e., a natural transformation of the form

$$\lambda : \Sigma(\mathrm{Id} \times (\mathcal{P}_f \mathcal{F}_{\mathfrak{W}})^A) \Longrightarrow (\mathcal{P}_f \mathcal{F}_{\mathfrak{W}} T^{\Sigma})^A \tag{4}$$

where A is the set of labels, $\mathfrak{W}$ is the commutative monoid of weights, $\Sigma = \coprod_{\mathtt{f}\in\Sigma} \mathrm{Id}^{\mathrm{ar}(\mathtt{f})}$ is the syntactic endofunctor induced by the process signature Σ, and T^Σ is the free monad for Σ. We will call natural transformations of this type *WF-GSOS distributive laws*.

Before stating the soundness theorem, we note that every natural transformation λ as above induces a $(\mathcal{P}_f\mathcal{F}_{\mathfrak{W}})^A$-coalgebra structure over ground Σ-terms. Namely, this is the only function $h_\lambda : T^\Sigma\emptyset \to (\mathcal{P}_f\mathcal{F}_{\mathfrak{W}}(T^\Sigma\emptyset))^A$ such that:

$$h_\lambda \circ a = (\mathcal{P}_f\mathcal{F}_{\mathfrak{W}}T^\Sigma(a^\#))^A \circ \lambda_X \circ \Sigma\langle id, h_\lambda\rangle \tag{5}$$

where $a^\# : T^\Sigma T^\Sigma\emptyset \to T^\Sigma\emptyset$ is the inductive extension of a.

We can now provide the soundness result for WF-GSOS specifications with respect to WF-GSOS distributive laws, and between systems and coalgebras they induce over ground Σ-terms.

Theorem 5.5 (Soundness). *A specification $\langle\mathcal{R}, \{\!|\text{-}|\!\}\rangle$ yields a natural transformation λ as in (4) such that h_λ and the ULTraS induced by $\langle\mathcal{R}, \{\!|\text{-}|\!\}\rangle$ coincide.*

Proof. For any set X, define the function λ_X as the composite:

$$\Sigma(X \times (\mathcal{P}_f\mathcal{F}_{\mathfrak{W}}X)^A) \xrightarrow{[\![\mathcal{R}]\!]_X} (\mathcal{P}_f T^\Theta(X + \mathcal{F}_{\mathfrak{W}}X))^A \xrightarrow{(\mu\circ\mathcal{P}_f\{\!|\text{-}|\!\}_X)^A} (\mathcal{P}_f\mathcal{F}_{\mathfrak{W}}T^\Sigma X)^A$$

where $\mu : \mathcal{P}_f\mathcal{P}_f \Rightarrow \mathcal{P}_f$ and $[\![\mathcal{R}]\!]_X$ is defined as follows: for all $\psi' \in T^\Theta(X + \mathcal{F}_{\mathfrak{W}}X)$, $\mathtt{f} \in \Sigma$, $c \in A$, trigger $\vec{A} = \langle A_1, \dots A_n\rangle$, $\vec{w} = \langle w_1, \dots w_p\rangle$, $y'_k \in X$ and $\Phi_i(a) = \{\phi^a_{ij} \in \mathcal{F}_{\mathfrak{W}}X \mid 1 \le j \le m^a_i\}$ for $n = \mathrm{ar}(\mathtt{f})$ and $i \in \{1, \dots, n\}$, let

$$\psi' \in [\![\mathcal{R}]\!]_X(\mathtt{f}((x'_1, \Phi_1), \dots, (x'_n, \Phi_n)))$$

if, and only if, there exists in $\mathcal{R}$ a (possibly renamed) rule

$$\frac{\left\{x_i \xrightarrow{a}\!\!\triangleright\ \phi^a_{ij}\right\}_{\substack{1\le i\le n,\\ a\in A_i,\\ 1\le j\le m^a_i}} \quad \left\{x_i \not\xrightarrow{b}\!\!\triangleright\ \right\}_{\substack{1\le i\le n,\\ b\in B_i}} \quad \left\{[\![\phi^{a_k}_{i_kj_k}]\!] = w_k\right\}_{1\le k\le p} \quad \left\{\phi^{a_k}_{i_kj_k}\big|^{\mathfrak{C}_k} \ni y_k\right\}_{1\le k\le q}}{\mathtt{f}(x_1, \dots, x_n) \xrightarrow{c}\!\!\triangleright\ \psi}$$

such that $m^a_i \neq 0$ iff $a \in A_i$ and there exists a substitution σ such that $\psi' = \sigma[\psi]$, $\sigma x_i = x'_i$, $\sigma y_k = y'_k$, $\sigma\phi^a_{ij} = \phi^a_{ij}$, $[\![\phi^{a_k}_{i_kj_k}]\!] = w_k$ and $\phi^{a_k}_{i_kj_k}(\sigma y_k) \in \mathfrak{C}_k$. Then, naturality can be proved separately for the two components: the former can be tackled as in [29, Th. 1.1] and the latter readily follows from Definition 3.3.

Correspondence of h_λ with the induced ULTraS follows by noting that the latter is given by structural recursion on Σ-terms by applying precisely λ as given above (cf. (5) and Definition 3.5). ◻

Now, by general results from the bialgebraic framework, every behavioural equivalence on h_λ is also a congruence on $T^\Sigma\emptyset$. In order to obtain this result we need the following (simple yet important) property.

Proposition 5.6. *The category of $(\mathcal{P}_f\mathcal{F}_{\mathfrak{W}})^A$-coalgebras has a final object.*

Proof. By [3] every finitary Set endofunctor admits a final coalgebra. By definition $\mathcal{F}_{\mathfrak{W}}$ is finitary. The thesis follows from $\mathcal{P}_f \cong \mathcal{F}_2$ and from finitarity being preserved by functor composition. ◻

Corollary 5.7 (Congruence). *Behavioural equivalence on the coalgebra over $T^\Sigma\emptyset$ induced by $\langle\mathcal{R}, \{\!|\text{-}|\!\}\rangle$ is a congruence with respect to the signature Σ.*

Proof. The syntactic endofunctor Σ admits an initial algebra and, by Proposition 5.6, the behavioural endofunctor $(\mathcal{P}_f\mathcal{F}_{\mathfrak{W}})^A$ admits a final coalgebra. The same holds for their free monad and cofree copointed functor respectively. The specification $\langle\mathcal{R}, \{\!|\text{-}|\!\}\rangle$ defines, by Theorem 5.5, a distributive law which uniquely extends to a distributive law distributing the free monad over the cofree copointed functor; then the thesis follows from [29, Cor. 7.3]. ◻

$$
\begin{tikzcd}
\Sigma(\mathrm{Id} \times (\mathcal{P}_f \mathcal{F}_{\mathfrak{W}})^A) \arrow[rrr, "\lambda"] \arrow[dr, "\rho"] \arrow[ddd, "\llbracket \mathcal{R} \rrbracket"'] & \arrow[r, phantom, "\text{(Lem. 5.9)}"] & & (\mathcal{P}_f \mathcal{F}_{\mathfrak{W}} T^{\Sigma})^A \\
& (\mathcal{P}_f T^{\Theta}(\mathrm{Id} + \mathcal{F}_{\mathfrak{W}}))^A \arrow[r, "(\mathcal{P}_f \theta)^A"] & (\mathcal{P}_f^2 \mathcal{F}_{\mathfrak{W}} T^{\Sigma})^A \arrow[ur, "(\mu_{\mathcal{F}_{\mathfrak{W}} T^{\Sigma}})^A"] & \\
& (\mathcal{P}_f^2 T^{\Theta}(\mathrm{Id} + \mathcal{F}_{\mathfrak{W}}))^A \arrow[u, "(\mu_{T^{\Theta}(\mathrm{Id}+\mathcal{F}_{\mathfrak{W}})})^A"] \arrow[r, "(\mathcal{P}_f^2 \theta)^A"'] \arrow[ur, phantom, "\text{(Nat.)}"] & (\mathcal{P}_f^3 \mathcal{F}_{\mathfrak{W}} T^{\Sigma})^A \arrow[u, "(\mathcal{P}_f \mu_{\mathcal{F}_{\mathfrak{W}} T^{\Sigma}})^A"] \arrow[dr, "(\mathcal{P}_f \mu_{\mathcal{F}_{\mathfrak{W}} T^{\Sigma}})^A"] \arrow[r, phantom, "(=)"] & \\
(\mathcal{P}_f T^{\Xi}(\mathrm{Id} + \mathcal{F}_{\mathfrak{W}}))^A \arrow[ur, "(\mathcal{P}_f \xi)^A"] \arrow[rrr, "(\mathcal{P}_f \{\!|\text{-}|\!\})^A"'] & \arrow[r, phantom, "\text{(Def. } \{\!|\text{-}|\!\}\text{)}"] & & (\mathcal{P}_f \mathcal{F}_{\mathfrak{W}} T^{\Sigma})^A \arrow[uuu, "(\mu_{\mathcal{F}_{\mathfrak{W}} T^{\Sigma}})^A"']
\end{tikzcd}
$$

Figure 2: Factorization for λ-distributive laws as WF-GSOS specifications.

5.4 WF-GSOS distributive laws are WF-GSOS specifications

In this subsection we give the important result that the WF-GSOS format is also *complete* with respect to distributive laws of the form (4).

Theorem 5.8 (Completeness)**.** *Every WF-GSOS distributive law λ arises from some WF-GSOS specification $\langle \mathcal{R}, \{\!|\text{-}|\!\} \rangle$.*

The proof of this Theorem follows the methodology introduced by Bartels for proving adequacy of Bloom's GSOS specification format [4, §3.3.1]. The (rather technical) proof will take the rest of this subsection, so for sake of conciseness we omit to recall some results which can be found in *loc. cit.*.

The thesis follows from proving that, for every λ, there exists an image-finite set of WF-SOS rules $\mathcal{R}$ (and suitable interpretations θ and ξ) making the diagram in Figure 2 commute. The lower part of the diagram defines the interpretation $\{\!|\text{-}|\!\}$ out of ξ and θ completing the WF-GSOS specification for λ. The middle and right parts of the diagram trivially commute.

The upper part of the diagram commutes because of the following lemma which states that every WF-GSOS distributive law arises from an interpretation and a natural transformation having the same type of those defined by image-finite sets of WF-GSOS rules.

Lemma 5.9. *Let Σ, A and $\mathfrak{W}$ be a signature, a set of labels and a commutative monoid, respectively. Let λ be a WF-GSOS distributive law as in* (4)*. There exist Θ and an interpretation factorizing λ i.e. there exists $\rho : \Sigma(\mathrm{Id} \times (\mathcal{P}_f \mathcal{F}_{\mathfrak{W}})^A) \Rightarrow (\mathcal{P}_f T^{\Theta}(\mathrm{Id} + \mathcal{F}_{\mathfrak{W}}))^A$ such that $\lambda = (\mu \circ \mathcal{P}_f \theta)^A \circ \rho$.*

Proof (sketch). In Set it is easy to encode finitely supported functions as terms. For instance let Θ extend Σ with operators for describing collections and weight assignments (e.g. $(\text{-} \mapsto w)$ where $w \in \mathfrak{W} \setminus \{0\}$). Then, we can turn λ into ρ by simply encoding its codomain. Then θ simply evaluates these terms back to weight functions everything else to the $\emptyset$. $\square$

Following Bartels' methodology, the left part of the diagram commutes by reducing ρ to simpler, but equivalent, families of natural transformations and eventually deriving a syntactical specification which is then shown to be equivalent to an image-finite set of WF-GSOS rules and an intermediate interpretation ξ. The use of another signature Ξ besides Θ gives us an extra degree of freedom and simplifies the proof. In particular, it allows us to encode natural transformations of type $\mathcal{F}_{\mathfrak{W}} \Rightarrow \mathcal{P}_f \mathcal{F}_{\mathfrak{W}}$ (yielded by the aforementioned reduction) in ξ and handle them downstream to the interpretation $\{\!|\text{-}|\!\}$. This expressiveness gain is one of the reasons for the introduction of non-determinism in Definition 3.3.

First, note that, by [4, Lem. A.1.1], ρ as above is equivalent to:

$$\bar{\rho} : \Sigma(\mathrm{Id} \times (\mathcal{P}_f \mathcal{F}_{\mathfrak{W}})^A) \times A \Longrightarrow \mathcal{P}_f T^{\Theta}(\mathrm{Id} + \mathcal{F}_{\mathfrak{W}})$$

which is equivalent to a family of natural transformations

$$\alpha_{\mathtt{f},c} : (\mathrm{Id} \times (\mathcal{P}_f \mathcal{F}_{\mathfrak{W}})^A)^N \Longrightarrow \mathcal{P}_f T^{\Theta}(\mathrm{Id} + \mathcal{F}_{\mathfrak{W}}) \tag{6}$$

indexed by $\mathtt{f} \in \Sigma$ and $c \in A$ and where $N = \{1, \dots, \mathrm{ar}(\mathtt{f})\}$. In fact, Σ is a polynomial functor and $\mathrm{Id} \times A \cong A \cdot \mathrm{Id}$ is an $|A|$-fold coproduct.

By [4, Lem. A.1.7], each $\alpha_{\mathtt{f},c}$ is equivalent to a natural transformation

$$\bar{\alpha}_{\mathtt{f},c} : (\mathcal{P}_f \mathcal{F}_{\mathfrak{W}})^{A \times N} \Longrightarrow \mathcal{P}_f T^{\Theta}(N + \mathrm{Id} + \mathcal{F}_{\mathfrak{W}}) \tag{7}$$

and, by the natural isomorphism

$$(\mathcal{P}_f)^{A \times N} \cong (\mathcal{P}_f^+ + 1)^{A \times N} \cong \coprod_{E \subseteq A \times N} (\mathcal{P}_f^+)^E$$

each $\bar{\alpha}_{\mathtt{f},c}$ is equivalent to a family of natural transformations

$$\beta_{\mathtt{f},c,E} : (\mathcal{P}_f^+ \mathcal{F}_{\mathfrak{W}})^E \Longrightarrow \mathcal{P}_f T^{\Theta}(N + \mathrm{Id} + \mathcal{F}_{\mathfrak{W}}) \tag{8}$$

where the added index corresponds to the vector of sets of labels $\langle E_1, \dots, E_{\mathrm{ar}(\mathtt{f})} \rangle$ composing the trigger of a WF-GSOS rule. By the natural isomorphism

$$\mathcal{P}_f^+ \mathcal{F}_{\mathfrak{W}} \cong \mathcal{P}_f^+ \coprod_{v \in \mathfrak{W}} \mathcal{F}_{\mathfrak{W}}^v \cong \coprod_{V \in \mathcal{P}_f^+ \mathfrak{W}} \prod_{v \in V} \mathcal{P}_f^+ \mathcal{F}_{\mathfrak{W}}^v$$

where $\mathcal{F}_{\mathfrak{W}}^v X \triangleq \{\phi \in \mathcal{F}_{\mathfrak{W}} X \mid [\![\phi]\!] = v\}$, each $\beta_{\mathtt{f},c,E}$ is equivalent to a family of natural transformations

$$\gamma_{\mathtt{f},c,E,w} : \coprod_{e \in E} \prod_{v \in w(e)} \mathcal{P}_f^+ \mathcal{F}_{\mathfrak{W}}^v \Longrightarrow \mathcal{P}_f T^{\Theta}(N + \mathrm{Id} + \mathcal{F}_{\mathfrak{W}}) \tag{9}$$

where $w : E \to \mathcal{P}_f^+ \mathfrak{W}$. Since total weight premises associate pairs from E to weights, maps like w can be seen as families of triggering weights.

By [4, Lem. A.1.3] and by the natural isomorphism

$$T^{\Theta} \cong \coprod_{\psi \in T^{\Theta} 1} \mathrm{Id}^{|\psi|_*}$$

where $|\psi|_*$ denotes the number of occurrences of $* \in 1$ in the Θ-term ψ (cf. [4, Lem. A.1.5]) each $\gamma_{\mathtt{f},c,E,w}$ corresponds to a family of natural transformations

$$\delta_{\mathtt{f},c,E,w,\psi} : \coprod_{e \in E} \prod_{v \in w(e)} \mathcal{P}_f^+ \mathcal{F}_{\mathfrak{W}}^v \Longrightarrow \mathcal{P}_f^+((\mathrm{Id} + \mathcal{F}_{\mathfrak{W}})^{|\psi|_*}) \tag{10}$$

where the added index ψ ranges over some subset of $T^{\Theta}(1 + N)$ (cf. target terms of WF-GSOS rules).

Then, following [4, §3.3.1, Cor. A.2.8] it is easy to check that each $\delta_{\mathtt{f},c,E,w,\psi}$ describes a non-empty, finite set of derivation rules as

$$\frac{\phi_{j,v_j} \in \pi_{v_j}(\Phi_{e_j}) \quad y_i \in \epsilon_{j,v_j}(\phi_{j,v_j})}{\langle z_1, \dots, z_{|\psi|_*} \rangle \in \delta_{\mathtt{f},c,E,w,\psi}((\Phi_e)_{e \in E})}$$

where $p, q \in \mathbb{N}$, $e_j \in E$, $1 \le j \le p$, $1 \le i \le q$, $v_j \in w(e_j)$, each $z_k \in \{y_i \mid 1 \le i \le q\}$ for $1 \le k \le |\psi|_*$ and each ϵ_{j,v_j} is a natural transformation:

$$\epsilon_{j,v_j} : \mathcal{F}_{\mathfrak{W}}^{v_j} \Longrightarrow \mathcal{P}_f^+(\mathrm{Id} + \mathcal{F}_{\mathfrak{W}}).$$

Natural transformations of this type can be easily encoded in the term ψ by suitable extensions of Θ and therefore each $\delta_{\mathtt{f},c,E,w,\psi}$ can be shown to be equivalent to a δ-specification i.e. a non-empty, finite set of derivation rules as above except for each z_k being a term wrapping $\phi_{j,v}$ with the symbol denoting $\epsilon_{j,v}$. These terms are then evaluated by the interpretation ξ^{δ} as expected.

This proof points out the trade-off that has to be made in presence of specifications with interpretation such as WF-GSOS or MGSOS [2]. In fact, clubs were not mentioned in the above reduction of ρ since each ϵ_{j,v_j} was handled by the interpretation ξ. However, the following result shows that clubs (hence, premises like $\phi\big|^{\mathfrak{C}} \ni y$), characterize natural transformations of type $\mathcal{F}^{v}_{\mathfrak{W}} \Rightarrow \mathcal{P}_f$.

Lemma 5.10. *For any natural transformation $\upsilon : \mathcal{F}^{w}_{\mathfrak{W}} \Rightarrow \mathcal{P}_f$ there exists a club $\mathfrak{C}_\upsilon$ characterizing it: $x \in \upsilon_X(\phi) \iff \phi(x) \in \mathfrak{C}_\upsilon$.*

Proof (sketch). Intuitively, natural transformations of this type are "selecting a finite subset from each weight function domain" and it is easy to check that elements can be only singled out by their weight. Likewise, finiteness and naturality prevent the selection of anything outside function supports. Then, the problem readily translates into finding the finest topology on the weight monoid that "plays well" with $\mathcal{F}_{\mathfrak{W}}$ i.e. such that monoidal addition, seen as a continuous map from the product topology, preserves opens (i.e. any admissible selection). Clubs are a base for this topology since, by definition, these are the only substructures isolated w.r.t. $\mathcal{F}_{\mathfrak{W}}$-action. Hence selections made by υ are completely characterized by a single club $\mathfrak{C}_\upsilon$. $\square$

Finally, we have to translate the set of rules we got so far into the WF-GSOS format; we do it by reversing the chain that led us from ρ to δ and δ-specification. By Lemma 5.10 every δ-specification is equivalent to a γ-specification

$$\left\{ \frac{\phi_j \in \Phi_{e_j} \quad \llbracket\phi_j\rrbracket = v_j \quad \phi_j\big|^{\mathfrak{C}_i} \ni y_i}{\psi(z_1,\dots,z_{|\psi|_*}) \in \gamma_{\mathtt{f},c,E,w}((\Phi_e)_{e\in E})} \;\middle|\; \begin{array}{l} v_j \in w(e_j), \\ z_k \in \{\phi_j[\zeta_j], y_i\} + N, \\ \psi \end{array} \right\}_{\text{fininte}}$$

where $\phi_j[\zeta_j]$ is a term build with the Ξ-operator denoting the natural transformation $\zeta_j : \mathcal{F}^{v_j}_{\mathfrak{W}} \Rightarrow \mathcal{P}_f\mathcal{F}_{\mathfrak{W}}$ and ξ^γ acts as ξ^δ on these terms, as the identity on those generated from Θ (distributing the powerset as expected) and maps everything else to $\emptyset$. A γ-specification defines a natural transformation as in (9) and every family of γ-specifications characterizing a natural transformation as in (8) is equivalent to a β-specification i.e. a set of derivation rules

$$\left\{ \frac{\phi_j \in \Phi_{e_j} \quad \llbracket\phi_j\rrbracket = v_j \quad \phi_j\big|^{\mathfrak{C}_i} \ni y_i}{\psi(z_1,\dots,z_{|\psi|_*}) \in \beta_{\mathtt{f},c,E}((\Phi_e)_{e\in E})} \;\middle|\; \begin{array}{l} z_k \in \{\phi_j[\zeta_j], y_i\} + N, \\ \psi \end{array} \right\}_{\text{image finite}}$$

finite up to vectors of total weights $\vec{v} = \langle v_0, \dots, v_p\rangle$. Since $E \subseteq A \times N$, every family of β-specifications describing a natural transformation as in (7) is equivalent to a set

$$\left\{ \frac{\Phi_{m_n,b_n} = \emptyset \quad \phi_j \in \Phi_{l_j,a_j} \quad \llbracket\phi_j\rrbracket = v_j \quad \phi_j\big|^{\mathfrak{C}_i} \ni y_i}{\psi(z_1,\dots,z_{|\psi|_*}) \in \bar{\alpha}_{\mathtt{f},c}(\langle \Phi_1,\dots,\Phi_{|\mathtt{f}|}\rangle)} \;\middle|\; \begin{array}{l} \langle m_n,b_n\rangle \neq \langle l_j,a_j\rangle, \\ z_k \in \{\phi_j[\zeta_j], y_i\} + N, \\ \psi \end{array} \right\}_{\text{im.fin.}}$$

containing finitely many rules for every E and $\vec{v}$. This set corresponds to an α-specification i.e. an image-finite set like the following:

$$\left\{ \frac{\Phi_{m_n}(b_n) = \emptyset \quad \phi_j \in \Phi_{l_j}(a_j) \quad \llbracket\phi_j\rrbracket = v_j \quad \phi_j\big|^{\mathfrak{C}_i} \ni y_i}{\psi(z_1,\dots,z_{|\psi|_*}) \in \alpha_{\mathtt{f},c}(\langle\langle x_1, \Phi_1\rangle,\dots,\langle x_{|\mathtt{f}|}, \Phi_{|\mathtt{f}|}\rangle\rangle)} \;\middle|\; \begin{array}{l} \langle m_n,b_n\rangle \neq \langle l_j,a_j\rangle, \\ z_k \in \{\phi_j[\zeta_j], y_i, x_h\}, \\ \psi \end{array} \right\}_{\text{im.fin.}}$$

Finally, every family of α-specifications equivalent to a natural transformation as ρ corresponds to an image-finite set of WF-GSOS rules and an interpretation. Therefore we conclude that for any ρ there exist $\mathcal{R}$ and ξ as in Figure 2.

6 Conclusions and future work

In this paper we have presented WF-GSOS, a GSOS-style format for specifying non-deterministic systems with quantitative aspects. A WF-GSOS specification is composed by a set of rules for the derivation of judgements of the form $P \xrightarrow{a}\!\!\rhd\ \psi$, where ψ is a term of a specific signature, together with an *interpretation* for these terms as weight functions. We have shown that a specification in this format defines an ULTraS, and it is expressive enough to subsume other more specific formats such as Klin's *Weighted GSOS* for WLTS [21], and Bartel's *Segala-GSOS* for Segala systems [4, §5.3], and those subsumed by them e.g. Klin and Sassone's Stochastic GSOS [21] and Bloom's GSOS [7]. WF-GSOS induces naturally a notion of (strong) bisimulation, which we have compared with $\mathcal{M}$-bisimulation used in ULTraS. We have also provided a general categorical presentation of ULTraSs as coalgebras of a precise class of functors, parametric on the underlying weight structure. This presentation allows us to define categorically the notion of *abstract GSOS* for ULTraS, i.e., natural transformations of a precise type. We have proved that WF-GSOS specification format is *adequate* (i.e., sound and complete) with respect to this notion. Taking advantage of Turi-Plotkin's bialgebraic framework, we have proved that the bisimulation induced by a WF-GSOS is always a congruence; hence our specifications can be used for compositional and modular reasoning in quantitative settings (e.g., for ensuring performance properties). Moreover, the format is at least as expressive as every GSOS specification format for systems subsumed by ULTraS.

Related works In this paper we have shown that commutative monoids are enough to define ULTraSs, their homomorphisms and bisimulations. The original work [5] assumed weights to be organised into a partial order with bottom $(W, \leq, \bot)$, but the order plays no rôle in the definition besides distinguishing the point $\bot$ used to express unreachability. A monoidal sum is eventually and implicitly assumed by the notion of $\mathcal{M}$-bisimulation and, because of the definition of M-function, this operation is assumed to be be monotone in both its components and to have $\bot$ as unit. In other words, $\mathcal{M}$-bisimulation implicitly assumes weights to form a commutative positively ordered monoid $(W, +, \leq, 0)$. Any such a monoid is positive and hence it has a natural order $a \trianglelefteq b \iff \exists c.\, a + c = b$; this order is the weakest one rendering the monoid W positively ordered, in the sense that for any such ordering $\leq$, it is $\trianglelefteq \subseteq \leq$.

We note that in [5], weights used to define ULTraSs are decoupled from those of M-functions; e.g., the formers can be in $([0,1], \leq, 0)$ and the latters in $(\mathbb{R}_0^+, +, 0)$. However, the notion of constrained ULTraS is sill needed to precisely capture probabilistic systems or, in other words, the use of partial orders may still require to embed the systems under study into a larger class of ULTraSs. We remark that $(\mathbb{R}_0^+, +, 0)$ is the smaller completion of $([0,1], \leq, 0)$ under $+$ and in this sense the embedding can be seen as canonical. Therefore, defining ULTraSs in terms of commutative monoids is a conservative generalisation that additionally provides a natural notion of homomorphisms and hence bisimulations. As a side note, existence of bottoms does not allow weights to have opposites, e.g., to model opposite transitions like in calculi for reversible computations.

Although in this paper we have taken ULTraSs as a reference, WF-GSOS can be interpreted in other meta-models, such as FuTSs [23]. Like ULTraSs, FuTSs have state-to-function transitions, but admit several distinct domains for weight functions and more free structure besides the strict alternation between non-deterministic and quantitative steps. In their more general form, they can be understood as coalgebras for functors of shape:

$$F_{\vec{A},\vec{\mathfrak{W}}} = (\mathcal{F}_{\mathfrak{W}_{0,k_0}} \dots \mathcal{F}_{\mathfrak{W}_{1,o}})^{A_0} \times \dots (\mathcal{F}_{\mathfrak{W}_{n,k_n}} \dots \mathcal{F}_{\mathfrak{W}_{1,o}})^{A_n} \tag{11}$$

where each $\mathfrak{W}_{i,j}$ in $\vec{\mathfrak{W}}$ is a commutative monoid and each A_i in $\vec{A}$ is a set. We remark that, although in [23] weights are drawn from semirings, commutative monoids are sufficient to define $\mathcal{F}_{\mathfrak{W}}$ and hence define FuTS, homomorphisms and eventually bisimulations. Moreover, Lemma 5.4

readily generalises to (11): if weights are drawn only from positive refinement monoids then any such functor is wpp. No rule format for FuTSs has been published yet; we believe the WF-GSOS specification format to be a step in this direction because of the similarities between the behavioural functors involved. This would allow us to formulate compositionality results for (meta)calculi defining FuTSs, e.g., the framework for stochastic calculi proposed in [13]. Indeed since ULTraSs can be viewed as FuTSs (assuming commutative monoids as a common ground) any specification format for the latter that is both correct and complete w.r.t. the suitable abstract GSOS law will necessarily subsume WF-GSOS.

The systems considered in this paper can be seen as generalised Segala systems. We showed how the proposed format subsumes Bartels' Segala-GSOS; however, this is not the only specification format for this kind of systems. In [11] Gebler et al. proposed a $nt\mu f\nu/nt\mu x\nu$ rule format for describing Segala systems. Since Turi-Plotkin seminal paper [29] it is well known that GSOS and coGSOS (i.e tree-rule formats such as that in [11]) correspond to distributive laws of completely different shapes: the former distribute monads over copointed endofunctors whereas the latter distribute pointed endofunctors over comonads. These different shapes have obvious implications on the data available to the derivation rules: monads provide views "inside terms" whereas comonads provide views "inside executions". Their common generalisation are laws distributing monads over comonads but has limited practical benefits because it does not translate to any concrete rule format that would be complete for any specification containing both GSOS and coGSOS [20].

Future work The categorical characterization of ULTraS systems paves the way for further interesting lines of research. One is to develop Hennessy-Milner style modal logics for quantitative systems at the generality level of the ULTraS framework. In fact, Klin has shown in [18] that HML and CCS are connected by a (contravariant) adjunction. A promising direction is to follow this connection taking advantage of the bialgebraic presentation of ULTraSs provided in this paper. Another is to explore the implications of the recent developments in the coalgebraic understanding of unobservable moves [9, 8] in the context of this work. An intermediate step in this direction is to develop a suitable monad structure for $\mathcal{P}_f\mathcal{F}_{\mathfrak{W}}$ which is, in general, not a monad (cf. $\mathcal{P}_f\mathcal{D}$ where $\mathcal{D}$ is the probability distribution monad). This alone will allow us to define e.g. trace and testing equivalences in a principled coalgebraic way.

Acknowledgements We thank Rocco De Nicola, Daniel Gebler, the anonymous reviewers and the QAPL'14 participants for useful discussions on the conference version of this paper. This work is partially supported by MIUR PRIN project 2010LHT4KM, *CINA*.

References

[1] P. Aczel and N. Mendler. A final coalgebra theorem. In D. H. Pitt, D. E. Rydeheard, P. Dybjer, A. M. Pitts, and A. Poigné, editors, *Proc. CTCS*, volume 389 of *LNCS Science*, pages 357–365, 1989. Springer.

[2] G. Bacci and M. Miculan. Structural operational semantics for continuous state stochastic transition systems. *J. CSS*, 81(5):834–858, 2015.

[3] M. Barr. Terminal coalgebras in well-founded set theory. *TCS*, 114(2):299–315, 1993.

[4] F. Bartels. *On Generalised Coinduction and Probabilistic Specification Formats*. PhD thesis, CWI, Amsterdam, 2004.

[5] M. Bernardo, R. De Nicola, and M. Loreti. A uniform framework for modeling nondeterministic, probabilistic, stochastic, or mixed processes and their behavioral equivalences. *Inf. Comput.*, 225:29–82, 2013.

[6] M. Bernardo and R. Gorrieri. A tutorial on EMPA: A theory of concurrent processes with nondeterminism, priorities, probabilities and time. *TCS*, 202(1-2):1–54, 1998.

[7] B. Bloom, S. Istrail, and A. R. Meyer. Bisimulation can't be traced. *J. ACM*, 42(1):232–268, 1995.

[8] F. Bonchi, S. Milius, A. Silva, and F. Zanasi. Killing epsilons with a dagger: A coalgebraic study of systems with algebraic label structure. *TCS*, 2015.

[9] T. Brengos, M. Miculan, and M. Peressotti. Behavioural equivalences for coalgebras with unobservable moves. *CoRR*, abs/1411.0090, 2014.

[10] L. Cardelli and R. Mardare. The measurable space of stochastic processes. In *Proc. QEST*, pages 171–180. IEEE Computer Society, 2010.

[11] M. David Lee, D. Gebler, and P. R. D'Argenio. Tree rules in probabilistic transition system specifications with negative and quantitative premises. In B. Luttik and M. A. Reniers, editors, *EXPRESS/SOS*, volume 89 of *EPTCS*, pages 115–130, 2012.

[12] R. De Nicola, D. Latella, M. Loreti, and M. Massink. Rate-based transition systems for stochastic process calculi. In *Proc ICALP*, pages 435–446. Springer, 2009.

[13] R. De Nicola, D. Latella, M. Loreti, and M. Massink. A uniform definition of stochastic process calculi. *J. ACM*, 46(1):5, 2013.

[14] M. Droste, W. Kuich, and H. Vogler. *Handbook of weighted automata*. Springer, 2009.

[15] H. P. Gumm and T. Schröder. Monoid-labeled transition systems. *ENTCS*, 44(1):185–204, 2001.

[16] J. Hillston. *A compositional approach to performance modelling*. Cambridge University Press, 1996.

[17] J. Hillston. Process algebras for quantitative analysis. In *LICS*, pages 239–248. IEEE Computer Society, 2005.

[18] B. Klin. Bialgebraic methods and modal logic in structural operational semantics. *Inf. Comput.*, 207(2):237–257, 2009.

[19] B. Klin. Bialgebras for structural operational semantics: An introduction. *TCS*, 412(38):5043–5069, 2011.

[20] B. Klin and B. Nachyla. Distributive laws and decidable properties of SOS specifications. In J. Borgström and S. Crafa, editors, *EXPRESS/SOS*, volume 160 of *EPTCS*, pages 79–93, 2014.

[21] B. Klin and V. Sassone. Structural operational semantics for stochastic and weighted transition systems. *Inf. Comput.*, 227:58–83, 2013.

[22] K. G. Larsen and A. Skou. Bisimulation through probabilistic testing. *Inf. Comput.*, 94(1):1–28, 1991.

[23] D. Latella, M. Massink, and E. P. de Vink. A definition scheme for quantitative bisimulation. In *Proc. QAPL*, 2015. To appear in EPCTS.

[24] M. Miculan and M. Peressotti. GSOS for non-deterministic processes with quantitative aspects. In N. Bertrand and L. Bortolussi, editors, *Proc. QAPL*, volume 154 of *EPCTS*, pages 17–33, 2014.

[25] C. Priami. Stochastic pi-calculus. *Comput. J.*, 38(7):578–589, 1995.

[26] J. J. M. M. Rutten. Universal coalgebra: a theory of systems. *TCS*, 249(1):3–80, 2000.

[27] R. Segala and N. A. Lynch. Probabilistic simulations for probabilistic processes. *Nord. J. Comput.*, 2(2):250–273, 1995.

[28] S. Staton. Relating coalgebraic notions of bisimulation. *LMCS*, 7(1), 2011.

[29] D. Turi and G. Plotkin. Towards a mathematical operational semantics. In *Proc. LICS*, pages 280–291. IEEE Computer Society Press, 1997.

A Weighted transition Systems

Weighted labelled transition systems (e.g. [14, 21]) are LTS whose transition are assigned a weight drawn from a commutative monoid $\mathfrak{W} = (W, 0, +)$. Henceforth we will write $\mathfrak{W}$-LTS for $\mathfrak{W}$-Weighted LTS or in general WLTS if no specific monoid is intended.

Definition A.1 ([21, Def. 2])**.** *Given a commutative monoid* $\mathfrak{W} = (W, +, 0)$*, a* $\mathfrak{W}$-weighted LTS *is a triple* (X, A, ρ) *where:*

- X *is a set of* states *(processes);*
- A *is a set of* labels *(actions);*
- $\rho : X \times A \times X \to W$ *is a* weight function*, mapping each triple of* $X \times A \times X$ *to a weight.*

(X, A, ρ) *is said to be* image-finite *iff for each* $x \in X$ *and* $a \in A$*, the set* $\{y \in X \mid \rho(x, a, y) \neq 0\}$ *is finite.*

It is well-known that, for suitable choices of $\mathfrak{W}$ and constraints, WLTS subsume several kind of systems such as:

- $(\{\mathtt{tt}, \mathtt{ff}\}, \vee, \mathtt{ff})$ for non-deterministic systems;
- $(\mathbb{R}_0^+, +, 0)$ for rated systems [21, 12] (e.g. CTMCs);
- $(\mathbb{R}_0^+, +, 0)$ and $\forall x \in X, a \in A$ $\sum_{y \in X} \rho(x, a, y) \in \{0, 1\}$ for generative (or fully) probabilistic systems;
- $(\mathbb{R}_0^+, +, 0)$ and $\forall x$ $\sum_{a \in A, y \in X} \mathrm{P}(x, a, y) \in \{0, 1\}$ for reactive probabilistic systems;
- $(\mathbb{R}_0^+, \max, 0)$ for "capabilities" (weights denotes the capabilities of a process and similar capabilities add up to a stronger one);
- etc.

Moreover, Klin defined in [21] a notion for WLTS (based on cocongruences) which uniformly instantiates to known bisimulations for systems expressible in the WLTS framework.

Definition A.2 ([21, Def. 4])**.** *Given two* $\mathfrak{W}$*-LTS* (X, A, ϕ) *and* (Y, A, ψ)*, a* $\mathfrak{W}$-bisimulation *is a relation* $R \subseteq X \times Y$ *s.t. for each pair* $(x, y) \subseteq X \times Y$*,* $(x, y) \in R$ *implies that for each* $a \in A$ *and each* (C, D) *of* $R^\star$*:*

$$\sum_{c \in C} \phi(x, a, c) = \sum_{c \in D} \psi(y, a, d).$$

WLTS are precisely functional ULTraS and, as stated in Proposition 2.7, every weighted bisimulation for a WLTS is a bisimulation for the corresponding functional ULTraS and vice versa.

Proof of Proposition 2.7. Trivially, there is a 1-1 correspondence between $\mathfrak{W}$-LTS and functional $\mathfrak{W}$-ULTraS. Then, Proposition 2.7 readily follows by observing that, for any given pair of $\mathfrak{W}$-LTS/ULTraS $(X, A, \rightarrowtriangle_X)$ $(Y, A, \rightarrowtriangle_Y)$, Definition 2.6 degenerates in Definition A.2 because for any $x \xrightarrow{a} \phi$ there is exactly one $y \xrightarrow{a} \phi$. $\square$

The coalgebraic understanding of WLTS makes the correspondence even more immediate. In fact, there exists a bijective map between $\mathfrak{W}$-LTS with labels in A and $(\mathcal{F}_{\mathfrak{W}})^A$-coalgebras (cf. [21, Prop. 8]) and every $\mathfrak{W}$-weighted bisimulation arise from a cocongruence (cf. [21, Prop. 9]). Then, consider the natural transformations $F : (\mathcal{F}_{\mathfrak{W}})^A \Longrightarrow (\mathcal{P}_f\mathcal{F}_{\mathfrak{W}})^A$ and $G = (\mathcal{P}_f\mathcal{F}_{\mathfrak{W}})^A \Longrightarrow (\mathcal{F}_{\mathfrak{W}})^A$:

$$F_X(\phi)(a) \triangleq \{\phi(a)\} \qquad G_X(\Phi)(a) \triangleq \lambda x : X. \sum_{\rho \in \Phi(a)} \rho(x)$$

which lift the $\mathfrak{W}$-LTS behaviour to ULTraS and back. These extends by composition to the functors, $\widetilde{F}$ and $\widetilde{G}$, between the categories of coalgebras for $(\mathcal{F}_{\mathfrak{W}})^A$ and $(\mathcal{P}_f\mathcal{F}_{\mathfrak{W}})^A$.

$$\begin{tikzcd} (\mathcal{F}_{\mathfrak{W}})^A\text{-}\mathsf{CoAlg} \arrow[d, bend right, "\widetilde{F}"'] \\ (\mathcal{P}_f\mathcal{F}_{\mathfrak{W}})^A\text{-}\mathsf{CoAlg} \arrow[u, bend right, "\widetilde{G}"'] \end{tikzcd}$$

The two are not adjoint but, the former is faithful and injective on objects whereas the latter is full and surjective on objects. Moreover $\widetilde{G}$ preserves the final coalgebra.

The natural transformations F and G give rise to the arrows G-F and F-G (pictured below) by pre- and post- composition and such that the first is injective and the second is surjective.

$$\begin{tikzcd} \mathsf{Nat}(\Sigma(\mathrm{Id} \times (\mathcal{F}_{\mathfrak{W}})^A, (\mathcal{F}_{\mathfrak{W}}T^{\Sigma})^A) \arrow[d, bend right, "G\text{-}F"'] \\ \mathsf{Nat}(\Sigma(\mathrm{Id} \times (\mathcal{P}_f\mathcal{F}_{\mathfrak{W}})^A, (\mathcal{P}_f\mathcal{F}_{\mathfrak{W}}T^{\Sigma})^A) \arrow[u, bend right, "F\text{-}G"'] \end{tikzcd}$$

The functors above prove that on the same monoid ULTraS are a strict superclass of WLTS. By a quick cardinality reasoning it is possible to extend the inclusion result to the case where the monoid is allowed to change. In fact, for any $\mathfrak{W} = (W, +, 0)$ s.t. $|W| > 1$ there is no monoid $\mathfrak{V} = (V, \cdot, 1)$ such that $2^{(|W|^x)} = |V|^x$. Unfortunately we cannot rule out the possibility of "determinizing" every ULTraS to some WLTS while preserving and reflecting behavioural equivalences.

B Segala Systems

In their general format, Segala systems [27] are state machines (originally introduced as automata) whose transitions can be pictured as being made of two steps belonging to two different behavioural aspects: the first sub-step is non-deterministic and the second one is probabilistic. The following definitions are taken from [27] with minor notational differences and by restricting to finite probability distributions (whereas the original definition is given to discrete at most countable probability spaces) for conciseness and uniformity with the restriction to image-finite systems made in the paper.

Definition B.1. *A* Segala system *is a triple* $(X, A, \multimap\!\!\rhd)$ *where:*

- X *is a set of* states *(processes);*

- A *is a set of* labels *(actions);*
- $\rightarrowtriangle \subseteq X \times A \times \mathcal{D}(X)$ *a* transition relation *between states and discrete probability spaces over pairs of labels and states.*

Definition B.2 ([27, Def. 14])**.** *Let $(X, A, \rightarrowtriangle_X)$ and $(Y, A, \rightarrowtriangle_Y)$ be two Segala systems. A bisimulation is a relation $R \subseteq X \times Y$ such that for each $(x, y) \in X \times Y$, $(x, y) \in R$ implies that for each $a \in A$ for each $x \xrightarrow{a}\!\!\triangleright\ \phi$ there is $y \xrightarrow{a}\!\!\triangleright\ \psi$ s.t. for each $(C, D) \in R^\star$ $\sum_{c\in C} \phi(c) = \sum_{d\in D} \psi(d)$ and symmetrically for y.*

Proof of Proposition 2.8. Clearly $\mathcal{D}X \subseteq \mathcal{F}_{\mathbb{R}_0^+}X$ and hence Segala systems are constrained ULTraS. Then, Proposition 2.8 readily follows by observing that the two notions coincide on the non-deterministic part and then on the summations over elements of R^* i.e. the extension of R $\square$

C Omitted proofs

Proof of Proposition 2.11. The function $\mathcal{M}$ is well-given because $\mathcal{M}(x, a, C) = \bot = \bot$ whenever $x \not\xrightarrow{a}\!\!\triangleright$ or, for each $x \xrightarrow{a}\!\!\triangleright\ \rho$, $\rho(C) = 0$, and $\mathcal{M}(x, a, C_1) = \mathcal{M}(y, a, C_1)$ and $\mathcal{M}(x, a, C_2) = \mathcal{M}(y, a, C_2)$ implies $\mathcal{M}(x, a, C_1 \cup C_2) = \mathcal{M}(y, a, C_1 \cup C_2)$ by definition of $R_{\mathfrak{W}}$.

By Definition 2.6, whenever $x \xrightarrow{a}\!\!\triangleright\ \phi$ then $y \xrightarrow{a}\!\!\triangleright\ \psi$ s.t. $\phi(C) = \psi(C)$ for each $C \in X/R$ i.e. $\phi R_{\mathfrak{W}} \psi$ and the symmetric case for y. Therefore $(x, y) \in R$ implies that $\Phi_{x,a} \triangleq \{[\phi]_{R_{\mathfrak{W}}} \mid x \xrightarrow{a}\!\!\triangleright\ \phi\}$ and $\Phi_{y,a} \triangleq \{[\psi]_{R_{\mathfrak{W}}} \mid y \xrightarrow{a}\!\!\triangleright\ \psi\}$ are equal for each $a \in A$. We can safely add $\bot$ to both $\Phi_{x,a}$ and $\Phi_{y,a}$ since, whenever both x and y terminate, they are either both stuck or both terminal. In fact, equality and inequality are preserved while adding $\bot$ since $\Phi_{x,a} = \emptyset \implies \bot \notin \Phi_{y,a}$ (and vice versa) by hypothesis. For each $C \in X/R$ ($x, y \in X$ and $a \in A$) let $\Psi_{x,a,C} \triangleq (\Phi_{x,a} \setminus \{[\rho]_{R_{\mathfrak{W}}} \mid \rho(C) = 0\}) \cup \{\bot\}$. Clearly $\Phi_{x,a} \cup = \bigcup_{C \in X/R} \Psi_{x,a,C}$ and if $(x, y) \in R$ then $\Psi_{x,a,C} = \Psi_{y,a,C}$. Complementarily, if $(x, y) \notin R$ then there exists some $\phi \in \Phi_{x,a}$ s.t. for no $\psi \in \Phi_{y,a}$ $\phi R_{\mathfrak{W}} \psi$ or vice versa; w.l.o.g. assume the former. Hence there exists $C \in X/R$ such that $\phi(C) \neq \psi(C)$ whence $\Psi_{x,a,C} \neq \Psi_{y,a,C}$. Finally, we conclude by $\mathcal{M}(x, a, C) = \Psi_{x,a,C}$ for each $x \in X$, $a \in A$ and $C \in X/R$. $\square$

Proof of Proposition 5.2. Let $\mathfrak{W} = (W, +, 0)$ be a commutative monoid and let $(X, A, \rightarrowtriangle_X)$, $(Y, A, \rightarrowtriangle_Y)$, (X, α) and (Y, β) be two ULTraS over $\mathfrak{W}$ and their corresponding coalgebras (Proposition 5.1). Recall that a function $f : X \to Y$ is a also coalgebra morphism $f : \alpha \to \beta$ iff, for each $x \in X$, and $a \in A$:

$$f(x) \xrightarrow{a}\!\!\triangleright_Y \psi \iff x \xrightarrow{a}\!\!\triangleright_X \phi \wedge \psi = \phi[f]$$

where $\phi[f]$ denotes the action of f on ϕ (i.e. the function $\lambda y : Y. \sum_{x\in f^{-1}(y)} \phi(x)$) and function equality is defined point-wise as usual. Firstly, we prove that if R is a kernel relation of some jointly epic cospan of coalgebra mophism from α and β then it is a bisimulation. Let the aforementioned cospan be $(X, \alpha) \xleftarrow{f} (Z, \gamma) \xrightarrow{g} (Y, \beta)$, $(Z, A, \rightarrowtriangle_Z)$ the ULTraS for γ and assume x and y such that $f(x) = g(y)$. By definition of coalgebra morphism, $f(x) = z$ implies:

$$x \xrightarrow{a}\!\!\triangleright_X \phi \iff z \xrightarrow{a}\!\!\triangleright_Z \rho = \phi[f] = \lambda c : Z \sum_{x\in f^{-1}(c)} \phi(x).$$

Likewise $g(y) = z$ implies:

$$y \xrightarrow{a}\!\!\triangleright_Y \psi \iff z \xrightarrow{a}\!\!\triangleright_Z \rho = \psi[g] = \lambda c : Z \sum_{y \in g^{-1}(c)} \psi(y).$$

Therefore $f(x) = g(y)$ implies:

$$x \xrightarrow{a}\!\!\triangleright_X \phi \implies y \xrightarrow{a}\!\!\triangleright_Y \psi \wedge \forall C \in Z.Z \sum_{x \in f^{-1}(C)} \phi(x) = Z \sum_{y \in g^{-1}(C)} \psi(y)$$
$$y \xrightarrow{a}\!\!\triangleright_Y \psi \implies x \xrightarrow{a}\!\!\triangleright_X \phi \wedge \forall C \in Z.Z \sum_{x \in f^{-1}(C)} \phi(x) = Z \sum_{y \in g^{-1}(C)} \psi(y)$$

Then, we conclude by noting that if R is the kernel of f, g there is a bijective correspondence between its equivalence classes and elements in Z since every class is in the image of f or g by the jointly epic assumption.

For the converse, given a bisimulation R for $(X, A, \rightarrow\!\!\triangleright_X)$ $(Y, A, \rightarrow\!\!\triangleright_Y)$ let Z be the set of the equivalence classes in R and consider the ULTraS $(Z, A, \rightarrow\!\!\triangleright_Z)$ defined as follows:

$$C \xrightarrow{a}\!\!\triangleright_Z \lambda D : Z. \sum_{x' \in D} \phi(x') \iff x \xrightarrow{a}_X \phi \wedge x \in C$$
$$C \xrightarrow{a}\!\!\triangleright_Z \lambda D : Z. \sum_{y' \in D} \psi(y') \iff y \xrightarrow{a}_Y \psi \wedge y \in C$$

The two statements are redundant since $x, y \in C \iff xRy$ and hence iff for every $x \xrightarrow{a}_X \phi$ there is $y \xrightarrow{a}_Y \psi$ s.t. $\phi \equiv_R \psi$ and vice versa. Finally, class membership defines a jointly epic coalgebra cospan from the coalgebras associated to $(X, A, \rightarrow\!\!\triangleright_X)$ and $(Y, A, \rightarrow\!\!\triangleright_Y)$ to the one associated to $(Z, A, \rightarrow\!\!\triangleright_Z)$ by simply mapping each $x \in X$ and each $y \in Y$ to its class. $\square$