University of Udine

Department of Mathematics and Computer Science

PREPRINT

Weak bisimulations for labelled transition systems weighted over semirings

Marino Miculan, Marco Peressotti

Weak bisimulations for labelled transition systems weighted over semirings

Marino Miculan   Marco Peressotti

Dept. of Mathematics and Computer Science, University of Udine, Italy

marino.miculan@uniud.it   marco.peressotti@uniud.it

October 10, 2013

Abstract

Weighted labelled transition systems are LTSs whose transitions are given weights drawn from a commutative monoid. WLTSs subsume a wide range of LTSs, providing a general notion of strong (weighted) bisimulation. In this paper we extend this framework towards other behavioural equivalences, by considering *semirings* of weights. Taking advantage of this extra structure, we introduce a general notion of *weak weighted bisimulation*. We show that weak weighted bisimulation coincides with the usual weak bisimulations in the cases of non-deterministic and fully-probabilistic systems; moreover, it naturally provides a definition of weak bisimulation also for kinds of LTSs where this notion is currently missing (such as, stochastic systems). Finally, we provide a categorical account of the coalgebraic construction of weak weighted bisimulation; this construction points out how to port our approach to other equivalences based on different notion of observability.

1   Introduction

Many extensions of labelled transition systems have been proposed for dealing with quantitative notions such as execution times, transition probabilities and stochastic rates; see *e.g.* [8, 7, 20, 21, 26, 35] among others. This ever-increasing plethora of variants has naturally pointed out the need for general mathematical frameworks, covering uniformly a wide range of cases, and offering general results and tools. As examples of these theories we mention UL-TRASs [7] and *weighted labelled transition systems* (WLTSs) [40, 25, 27]. In particular, in a WLTS every transition is associated with a *weight* drawn from a commutative monoid $\mathfrak{W}$; the monoid structure defines how weights of alternative transitions combine. As we will recall in Section 2, by suitably choosing this monoid we can recover ordinary non-deterministic LTSs, probabilistic transition systems, and stochastic transition systems, among others. WLTSs offer a notion of *(strong) $\mathfrak{W}$-weighted bisimulation*, which can be readily instantiated to particular cases obtaining precisely the well-known Milner's strong bisimulation [32], Larsen-Skou's strong probabilistic bisimulation [28], strong stochastic bisimulation [21], *etc.*

However, in many situations strong bisimulations are too fine, and many coarser relations have been introduced since then. Basically, these *observational* equivalences do not distinguish systems differing only for unobservable or not relevant transitions. Likely the most widely known of these observational equivalences is Milner's *weak bisimulation* for non-deterministic LTSs [32] (but see [41, 42] for many variations). Weak bisimulations focus on systems' interactions (communications, synchronizations, etc.), ignoring transitions associated with systems' internal operations, hence called *silent* (and denoted by the τ).

Unfortunately, weak bisimulations become quite more problematic in models for stochastic systems, probabilistic systems, *etc.* The conundrum is that we do not want to observe τ-transitions but at the same time their quantitative effects (delays, probability distributions) are still observable and hence cannot be ignored. In fact, for quantitative systems there is no general agreement of what a weak bisimulation should be. As an example, consider the

1

stochastic system S_1 executing an action a at rate r, and a system S_2 executing τ at rate r_1, followed by an a at rate r_2: should these two systems be considered weakly bisimilar?

$$S_1 \quad \circ \xrightarrow{\quad a, r \quad} \triangleright \circ \qquad\qquad S_2 \quad \circ \xrightarrow{\quad \tau, r_1 \quad} \triangleright \circ \xrightarrow{\quad a, r_2 \quad} \triangleright \circ$$

Some approaches restrict to instantaneous τ-actions (and hence $r_2 = r$) [6]; others require that the average times of a's executions are the same the two systems - but still these can be distinguished by looking at the variances [5]. Therefore, it is not surprising that many definitions proposed in literature are rather *ad-hoc*, and that a general mathematical theory is still missing.

This is the problem we aim to address in this paper. More precisely, in Section 3 we introduce the uniform notion of *weak weighted bisimulation* which applies to labelled transition systems weighted over a *semiring*. The multiplication operation of semirings allows us to compositionally extend weights to multi-step transitions and traces. In Section 4 we show that our notion of weak bisimulation coincides with the known ones in the cases of non-deterministic and fully probabilistic systems, just by changing the underlying semiring. Moreover it naturally applies to stochastic systems, providing an effective notion of *weak stochastic bisimulation*. As a side result we introduce a new semiring of *stochastic variables* which generalizes that of rated transition systems [26].

Then, in Section 5 we present the general algorithm for computing weak weighted bisimulation equivalence classes, parametric in the underlying semiring. This algorithm is a variant of Kanellakis-Smolka's algorithm for deciding strong non-deterministic bisimulation [23]. Our solution builds on the refinement technique used for the *coarsest stable partition*, but instead of "strong" transitions in the original system we consider "weakened" ones. We prove that this algorithm is correct, provided the semiring satisfies some mild conditions, *i.e.* it is ω-complete. Finally, we discuss also its complexity, which is comparable with Kanellakis-Smolka's algorithm. Thus, this algorithm can be used in the verification of many kinds of systems, just by replacing the underlying semiring (boolean, probabilistic, stochastic, tropical, arctic, ...) and taking advantage of existing software packages for linear algebras over semirings.

In Section 6 we give a brief categorical account of weak weighted bisimulations. These will be characterized as cocongruences between suitably *saturated* systems, akin to the elegant construction of ϵ-elimination given in [36].

In Section 7 we give some final remarks and directions for further work.

2 Weighted labelled transition systems

In this section we recall the notion of *labelled transition systems weighted over a commutative monoid*, showing how these subsume non-deterministic, stochastic and probabilistic systems, among many others. Weighted LTSs were originally introduced by Klin in [25] as the prosecution and generalization of the work on stochastic SOS presented in [26] with Sassone and were further developed in [27].

In the following let $\mathfrak{W}$ denote a generic commutative (aka *abelian*) monoid $(W, +, 0)$, *i.e.* a set equipped with a distinguished element 0 and a binary operation $+$ which is associative, commutative and has 0 as left and right unit.

Definition 1 ($\mathfrak{W}$-LTS [25, Def. 1]). *Given a commutative monoid* $\mathfrak{W} = (W, +, 0)$, *a* $\mathfrak{W}$-*weighted labelled transistion system is a triple* (X, A, ρ) *where:*

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

- A *is an at most countable set of* labels*;*

- $\rho: X \times A \times X \to W$ *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 *(resp.* countable*) iff for each* $x \in X$ *and* $a \in A$, *the set* $\{y \in X \mid \rho(x, a, y) \neq 0\}$ *is finite (resp. countable). A state* $x \in X$ *is said* terminal *iff for every* $a \in A$ *and* $y \in X$: $\rho(x, a, y) = 0$.

For adherence to the notation used in [25] and to support the intuitions based on classical labelled transition systems we shall often write $\rho(x \xrightarrow{a} y)$ for $\rho(x, a, y)$; moreover, following a common notation for stochastic and probabilistic systems, we will write also $x \xrightarrow{a,w} y$ to denote $\rho(x, a, y) = w$.

The monoidal structure was not used in Definition 1 but for the existence of a distinguished element required by the image finiteness (resp. countability) property. The commutative monoidal structure of weights comes into play in the notion of *bisimulation*, where weights of transitions with the same labels have to be "summed". This operation is commonplace for stochastic LTSs, but at first it may appear confusing with respect to the notion of bisimulation of non-deterministic LTSs; we will explain it in Section 2.1.

Definition 2 (Strong $\mathfrak{W}$-bisimulation [25, Def. 3]). *Given a $\mathfrak{W}$-LTS (X, A, ρ), a (strong) $\mathfrak{W}$-bisimulation is an equivalence relation R on X such that for each pair (x, x') of elements of X, $(x, x') \in R$ implies that for each label $a \in A$ and each equivalence class C of R:*

$$\sum_{y \in C} \rho(x \xrightarrow{a} y) = \sum_{y \in C} \rho(x' \xrightarrow{a} y).$$

Processes x and x' are said to be $\mathfrak{W}$-bisimilar (or just bisimilar when $\mathfrak{W}$ is understood) if there exists a $\mathfrak{W}$-bisimulation $\sim_{\mathfrak{W}}$ such that $x \sim_{\mathfrak{W}} x'$.

Clearly $\mathfrak{W}$-bisimulations are closed under arbitrary unions ensuring the $\mathfrak{W}$-*bisimilarity* on any $\mathfrak{W}$-LTS to be the largest $\mathfrak{W}$-bisimulation over it.[1]

Remark 1. *In order for the above definition to be well-given, summations need to be well-defined. Intuitively this means that the $\mathfrak{W}$-LTS (X, A, ρ) does not exceed the expressiveness of its underlying monoid of weights $\mathfrak{W}$. Reworded, the system has to be image finite if the monoid admits only finite summations; image countable if the monoid admits countable summations, and so on.*

In [25, 27], for the sake of simplicity the authors restrict themselves to image finite systems (which is not unusual in the coalgebraic setting). In the present paper we extend their definitions to the case of countable images. This generalization allows to capture a wider range of systems and is crucial for the definition of weak and delay bisimulations.

In practice, Remark 1 is not a severe restriction, since the commutative monoids relevant for most systems of interest admit summations over countable sets. To supports this claim, in the rest of this Section we illustrate how non-deterministic, stochastic and probabilistic labelled transition systems can be recovered as systems weighted over commutative monoids whit countable sums. These kind of commutative monoids are often called *commutative ω-monoids*[2].

2.1 Non-deterministic systems are WLTS

This section illustrates how non-deterministic labelled transition systems [32] can be recovered as systems weighted over the commutative ω-monoid of logical values equipped with logical disjunction $\mathsf{2} \triangleq (\{\mathtt{tt}, \mathtt{ff}\}, \vee, \mathtt{ff})$.

Definition 3 (Non-deterministic LTS). *A non-deterministic labelled transition system is a triple $(X, A, \rightarrow)$ where:*

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

- *A is an at most countable set of* labels *(actions);*

- *$\rightarrow \subseteq X \times A \times X$ is the* transition relation.

---

[1] Actually, strong $\mathfrak{W}$-bisimulation has been proven to be a strong bisimulation in coalgebraic sense [25].
[2] Monoids can be readily extended to ω-monoids adding either colimits freely or an "∞" element.

As usual, we shall denote an a-labelled transition from x to y i.e. $(x, a, y) \in \rightarrow$ by $x \xrightarrow{a} y$. A state y is called successor of a given state x iff $x \xrightarrow{a} y$. If x has no successors then it is said to be terminal. If every state has a finite set of successors then the system is said to be image finite. Likewise it is said to be image countable if each state has at most countably many successors.

Every 2-valued weight function is a predicate defining a subset of its domain, turning $\rho : X \times A \times X \rightarrow 2$ equivalent to the classical definition of the transition relation $\rightarrow \subseteq X \times A \times X$.

Definition 4 (Strong non-deterministic bisimulation). *Let $(X, A, \rightarrow)$ be an LTS. An equivalence relation $R \subseteq X \times X$ is a (strong non-deterministic) bisimulation on $(X, A, \rightarrow)$ iff for each pair of states $(x, x') \in R$, for any label $a \in A$ and each equivalence class $C \in X/R$:*

$$\exists y \in C. x \xrightarrow{a} y \iff \exists y' \in C. x' \xrightarrow{a} y'.$$

Two states x and x' are said bisimilar *iff there exists a bisimulation relation $\sim_l$ such that $x \sim_l y$. The greatest bisimulation for $(X, A, \rightarrow)$ uniquely exists and is called* (strong) bisimilarity.

Strong 2-bisimulation and strong non-deterministic bisimulation coincide, since logical disjunction over the states in a given class C encodes the ability to reach C making an a-labelled transition.

2.2 Stochastic systems are WLTS

Stochastic systems have important application especially in the field of quantitative analysis, and several tools and formalisms to describe and study them have been proposed (*e.g.* PEPA [21], EMPA [8] and the stochastic π-calculus [35]). Recently, *rated transition systems* [26, 27, 33, 7] emerged as a convenient presentation of these kind of systems.

Definition 5 (Rated LTS [26, Sec. 2.2]). *A rated labelled transition system is a triple (X, A, ρ) where:*

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

- A *is a countable set of* labels *(actions);*

- $\rho : X \times A \times X \rightarrow \mathbb{R}_0^+$ *is the* rate function.

Semantics of stochastic processes is usually given by means of labelled continuous time Markov chains (CTMC). The real number $\rho(x, a, y)$ is interpreted as the parameter of an exponential probability distribution governing the duration of the transition from state x to y by means of an a-labelled action and hence encodes the underlying CTMC (for more information about CTMCs and their presentation by transition rates see *e.g.* [20, 21, 34, 35]).

Definition 6 (Strong stochastic bisimulation). *Given a rated system (X, A, ρ) an equivalence relation $R \subseteq X \times X$ is a (strong stochastic) bisimulation on (X, A, ρ) (or strong equivalence [21]) iff for each pair of states $(x, x') \in R$, for any label $a \in A$ and each equivalence class $C \in X/R$:*

$$\sum_{y \in C} \rho(x, a, y) = \sum_{y \in C} \rho(x', a, y).$$

Two states x and x' are said bisimilar *iff there exists a bisimulation relation $\sim_s$ such that $x \sim_s y$. The greatest bisimulation for $(X, A, \rightarrow)$ uniquely exists and is called* (strong) bisimilarity.

Rated transition systems (hence stochastic systems) are precisely WLTS weighted over the commutative monoid of nonnegative real numbers (closed with infinity) under addition $(\overline{\mathbb{R}}_0^+, +, 0)$ and stochastic bisimulations correspond to $\overline{\mathbb{R}}_0^+$-bisimulations, as shown in [25]. Moreover, $\overline{\mathbb{R}}_0^+$ is an ω-monoid since non-negative real numbers admit sums over countable families. In particular, the sum of a given countable family $\{x_i \mid i \in I\}$ is defined as the supremum of the set of sums over its finite subfamilies:

$$\sum_{i \in I} x_i \triangleq \sup \left\{ \sum_{i \in J} x_i \mid J \subseteq I, |J| < \omega \right\}.$$

2.3 Probabilistic systems are (Constrained) WLTS

This section illustrates how probabilistic LTSs are captured by weighted ones. We focus on fully probabilistic systems (also known as *generative systems*) [14, 28, 4] but in the end we provide some hints on other types of probabilistic systems.

Fully probabilistic system can be regarded as a specializations of non-deterministic transition systems where probabilities are used to resolve nondeterminism. From a slightly different point of view, they can also be interpreted as labelled Markov chains with discrete parameter set [24].

Definition 7 (Fully probabilistic LTS). *A fully probabilistic labelled transition system is a triple* (X, A, P) *where:*

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

(2) A *is a countable set of* labels *(actions);*

(3) $\mathrm{P} : X \times A \times X \to [0, 1]$ *is a function such that for any* $x \in X$ $\mathrm{P}(x, \_, \_)$ *is either a discrete probability measures for* $A \times X$ *or the constantly* 0 *function.*

In "reactive" probabilistic systems, in contrast to fully probabilistic systems, transition probability distributions are dependent on the occurrences of actions *i.e.* for any $x \in X$ and $a \in A$ $\mathrm{P}(x, a, \_)$ is either a discrete probability measures for X or the constantly 0 function.

Strong probabilistic bisimulation has been originally introduced by Larsen and Skou [28] for reactive systems and has been reformulated by van Glabbeek et al. [14] for fully probabilistic systems.

Definition 8 (Strong probabilistic bisimilarity). *Let* (X, A, P) *be a fully probabilistic system. An equivalence relation* $R \subseteq X \times X$ *is a* (strong probabilistic) *bisimulation on* (X, A, P) *iff for each pair of states* $(x, x') \in R$, *for any label* $a \in A$ *and any equivalence class* $C \in X/R$:

$$\mathrm{P}(x, a, C) = \mathrm{P}(x', a, C)$$

where $\mathrm{P}(x, a, C) \triangleq \sum_{y \in C} \mathrm{P}(x, a, y)$.

Two states x *and* x' *are said* bisimilar *iff there exists a bisimulation relation* $\sim_p$ *such that* $x \sim_p y$. *The greatest bisimulation for* (X, A, P) *uniquely exists and is called* bisimilarity.

It would be tempting to recover fully probabilistic systems as LTS weighted over the probabilities interval $[0, 1]$ but unfortunately the addition does not define a monoid on $[0, 1]$ since it is not a total operation when restricted $[0, 1]$. There exist various commutative monoids over the probabilities interval, leading to different interpretations of probabilistic systems (as will be shown in Section 4.4), but since in Definition 8 we sum probabilities of outgoing transitions (*e.g.* to compute the probability of reaching a certain set of states), the real number addition has to be used.

Remark 2 (On partial commutative monoids). *The theory of weighted labelled transition systems can be extended to consider partial commutative monoids (i.e.* $a + b$ *may be undefined but when it is defined then also* $b + a$ *is and commutativity holds) or commutative* σ-*monoids to handle sums over opportune countable families (thus relaxing the requirement of weights forming* ω-*monoids). However, every* σ-*semiring can be turned into an* ω-*complete one by adding a distinguished* $+\infty$ *element and resolving partiality accordingly.*

Klin [25] suggested to consider probabilistic systems as systems weighted over $(\mathbb{R}_0^+, +, 0)$ but subject to suitable constraints ensuring that the weight function is a state-indexed probability distribution and thus satisfies Definition 7. These *constrained* WLTSs were proposed to deal with reactive probabilistic systems.

Definition 9 (constrained $\mathfrak{W}$-LTS). *Let* $\mathfrak{W}$ *be a commutative monoid and* $\mathcal{C}$ *be a constraint family. A* $\mathcal{C}$-*constrained* $\mathfrak{W}$-*weighted labelled transistion system is a* $\mathfrak{W}$-*LTS* (X, A, ρ) *such that its weight function* ρ *satisfies the constraints* $\mathcal{C}$ *over* $\mathfrak{W}$.

Then, fully probabilistic labelled transition systems are precisely constrained $\mathbb{R}_0^+$-LTSs (X, A, ρ) subject to the constraint family:

$$\sum_{a \in A, y \in X} \rho(x, a, y) \in \{0, 1\} \text{ for } x \in X.$$

Likewise, reactive probabilistic systems are $\mathbb{R}_0^+$-LTSs subject to the constraint family:

$$\sum_{y \in X} \rho(x, a, y) \in \{0, 1\} \text{ for } x \in X \text{ and } a \in A.$$

Therefore strong bisimulations for these kind of systems are exactly strong $\mathbb{R}_0^+$-bisimulations.

3 Weak bisimulations for WLTS over semirings

In the previous section we illustrated how weighted labelled transitions systems can uniformly express several kinds of systems such as non-deterministic, stochastic and probabilistic systems. Remarkably, bisimulations for these systems were proved to be instances of weighted bisimulations.

In this section we show how other observational equivalences can be stated at the general level of the weighted transition system offering a treatment for these notions uniform across the wide range of systems captured by weighted ones. Due to space constraints we focus on weak bisimulation but eventually we discuss briefly how the proposed results can cover other notions of observational equivalence.

3.1 From transitions to execution paths

Let $(X, A + \{\tau\}, \rho)$ be a $\mathfrak{W}$-LTS. A *finite execution path* π for this system is a sequence of transition *i.e.* an alternating sequence of states and labels like

$$\pi = x_0 \xrightarrow{a_1} x_1 \xrightarrow{a_2} x_2 \ldots x_{n-1} \xrightarrow{a_n} x_n$$

such that for each transition $x_{i-1} \xrightarrow{a_i} x_i$ in the path:

$$\rho(x_{i-1} \xrightarrow{a_i} x_i) \neq 0.$$

Let π denote the above path, then set:

$$\text{length}(\pi) = n \qquad \text{first}(\pi) = x_0 \qquad \text{last}(\pi) = x_n \qquad \text{trace}(\pi) = a_1 a_2 \ldots a_n.$$

to denote the length, starting state, ending state and trace of π respectively.

In order to extend the definition of the weight function ρ to executions we need some additional structure on the domain of weights, allowing us to capture concatenation of transition. To this end, we require weights to be drawn from a semiring, akin to the theory of weighted automata. Recall that a semiring is a set W equipped with two binary operations $+$ and $\cdot$ called *addition* and *multiplication* respectively and such that:

- $(W, +, 0)$ is a commutative monoid and $(W, \cdot, 1)$ is a monoid;

- multiplication left and right distributes over addition:

$$a \cdot (b + c) = (a \cdot b) + (a \cdot c) \qquad (a + b) \cdot c = (a \cdot c) + (b \cdot c)$$

- multiplication by 0 annihilates W:

$$0 \cdot a = 0 = a \cdot 0.$$

Basically, the idea is to express parallel and subsequent transitions (*i.e.* branching and composition) by means of addition and multiplication respectively. Therefore, multiplication is not required to be commutative (*cf.* the semiring of formal languages). Distributivity ensures that execution paths are independent from the alternative branching *i.e.* given two executions sharing some sub-path, we are not interested in which is the origin of the sharing; as the following diagram illustrates:

$$
\begin{array}{ccc}
\bullet & & \\
\downarrow a & & \\
b \swarrow \bullet \searrow c & \equiv & a \nearrow \bullet \nwarrow a \\
\bullet \qquad \bullet & & b \downarrow \bullet \qquad \bullet \downarrow c \\
& & \bullet \qquad \bullet
\end{array}
\qquad
\begin{array}{ccc}
a \searrow \bullet \swarrow b & \equiv & a \downarrow \bullet \qquad \bullet \downarrow b \\
\downarrow c & & c \searrow \bullet \swarrow c \\
\bullet & & \bullet
\end{array}
\tag{1}
$$

Finally, since weights of (proper) transitions are always different from 0, the annihilation property means that no proper execution can contain improper transitions.

Then, the weight function ρ extends to finite paths by semiring multiplication (therefore we shall use the same symbol):

$$
\rho(x_0 \xrightarrow{a_1} x_1 \ldots \xrightarrow{a_n} x_n) \triangleq \prod_{i=1}^{n} \rho(x_{i-1}, a_i, x_i)
$$

In the following let $\mathfrak{W}$ be a semiring $(W, +, 0, \cdot, 1)$.

Semirings offer enough structure to extend weight function to finite execution paths compositionally but executions can also be (countably) infinite. Likewise countable branchings (*cf.* Remark 1), paths of countable length can be treated requiring multiplication to be defined also over (suitable) countable families of weights and obviously respect the semiring structure. However, the additional requirement for $(W, \cdot, 1)$ can be avoided by dealing with suitable sets of paths as long as these convey enough information for the notion of weak bisimulation (and observational equivalence in general). In particular, a finite path π determines a set of paths (possibly infinite) starting with π, thus π can be seen as a representative for the set. Moreover, the behavior of a system can be reduced to its *complete* executions: a path is called *complete* (or "full" [4]) if it is either infinite or ends in a *terminal* state.

Intuitively, we distinguish complete paths only up to the chosen representatives: longer representative may generate smaller sets of paths, and this can be thought in "observing more" the system. If two complete paths are distinguishable, we have to be able to distinguish them in a finite way *i.e.* there must be two representative with enough information to tell one set from the other. Otherwise, if no such representative exist, then the given complete paths are indeed equivalent. Therefore, it is enough to be able to compositionally weight (finite) representatives in order to distinguish any complete path.

The remaining of the subsection elaborates the above intuition defining a σ-algebra over complete paths (for each state). The method presented is a generalization to semirings of the one used in [2]. This structure allows to deal with sets of finite paths avoiding redundancies (*cf.* Example 3) and define weights compositionally.

Let $\mathrm{Paths}(x)$, $\mathrm{CPaths}(x)$ and $\mathrm{FPaths}(x)$ denote the sets of all, complete and finite paths starting in the state $x \in X$ respectively. Likewise, we shall denote the corresponding sets of paths *w.r.t.* any starting state as Paths, CPaths and FPaths respectively (*e.g.* $\mathrm{Paths} = \cup_{x \in X} \mathrm{Paths}(x)$). Paths naturally organize into a preorder by the prefix relation. In particular, given $\pi, \pi' \in \mathrm{Paths}(x)$ define $\pi \preceq \pi'$ if and only if one of the following holds:

1. $\pi \equiv x \xrightarrow{a_1} x_1 \ldots \xrightarrow{a_n} x_n$ and $\pi' \equiv x \xrightarrow{a'_1} x'_1 \ldots \xrightarrow{a'_n} x'_{n'}$ (both finite), $x_i = x'_i$ and $a_i = a'_i$ for $i \leq n \leq n'$;

2. $\pi \equiv x \xrightarrow{a_1} x_1 \ldots \xrightarrow{a_n} x_n$ and $\pi' \equiv x \xrightarrow{a'_1} x'_1 \ldots$ (one finite and the other infinite), $x_i = x'_i$ and $a_i = a'_i$ for $i \leq n$;

3. $\pi = \pi'$ (both infinite).

For each finite path $\pi \in \mathrm{FPaths}(x)$ define the *cone of complete paths generated by* π as follows:

$$\pi{\uparrow} \triangleq \{\pi' \in \mathrm{CPaths}(x) \mid \pi \preceq \pi'\}.$$

Cones are precisely the sets we were sketching in the intuition above and form a subset of the parts of $\mathrm{CPaths}(x)$:

$$\Gamma \triangleq \{\pi{\uparrow} \mid \pi \in \mathrm{FPaths}(x)\}.$$

This set is at most countable since the set $\mathrm{FPaths}(x)$ is so and every two of its elements are either disjoint or one the subset of the other as the following Lemmas state.

Lemma 1. *For any state $x \in X$, the set of finite paths $\mathrm{FPaths}(x)$ of an image countable $\mathfrak{W}$-LTS is at most countable.*

Proof. By induction on the length k of paths in $\mathrm{FPaths}(x)$, these are at most countable. In fact, for $k = 0$ there is exactly one path, ε and, taken the set of paths of length k be at most countable, then the set of those with length $k + 1$ is at most countable because the system is assumed to be image countable. Then $\mathrm{FPaths}(x)$ is at most countable since it is the disjoint union of

$$\{\pi \in \mathrm{Paths}(x) \mid \mathrm{length}(\pi) = k\}$$

for $k \in \mathbb{N}$. $\qquad\qquad\square$

Lemma 2. *Two cones $\pi{\uparrow}_1$ and $\pi_2{\uparrow}$ are either disjoint or one the subset of the other.*

Proof. For any $\pi \in \mathrm{CPaths}(x)$, we have by definition:

$$\pi \in \pi_1{\uparrow} \iff \pi_1 \preceq \pi \text{ and } \pi \in \pi_2{\uparrow} \iff \pi_2 \preceq$$

Then, if $\pi_1 \preceq \pi_2$ then $\pi \in \pi_2{\uparrow} \Rightarrow \pi \in \pi_1{\uparrow}$ (likewise for $\pi_2 \preceq \pi_1$). For the other case, since $\pi_1 \npreceq \pi_2 \wedge \pi_2 \npreceq \pi_1$ there is no π such that $\pi_1 \preceq \pi \wedge \pi_2 \preceq \pi$. $\qquad\square$

Given $\Pi \subseteq \mathrm{FPaths}(x)$, the set of all cones generated by its elements is denoted by $\Pi{\uparrow}$ and defined as the (at most countable) union of the cones generated by each $\pi \in \Pi$. If this union is over disjoint cones then Π is said to be *minimal*.

Minimality is not preserved by set union even if operands are disjoint and both minimal. As a counter example consider the sets $\{\pi\}$ and $\{\pi'\}$ for $\pi \prec \pi' \in \mathrm{FPaths}(x)$; both are minimal and disjoint, but their union is not minimal since $\pi'{\uparrow} \subseteq \pi{\uparrow}$. However, Π always has at least a subset Π' being minimal and such that

$$\Pi{\uparrow} = \Pi'{\uparrow}. \tag{2}$$

and among these there exists exactly one which is also minimal in the sense of prefixes:

Lemma 3. *For $\Pi \subseteq \mathrm{FPaths}(x)$, there exists a minimal subset $\Pi' \subseteq \Pi$ which satisfies (2), i.e. for any $\Pi'' \subseteq \Pi$ satisfying (2) we have: $\forall \pi'' \in \Pi'' \exists \pi' \in \Pi.\pi' \preceq \pi''$. We denote such Π' by $\Pi{\downarrow}$.*

Proof. Clearly $\Pi{\uparrow} = \emptyset$ iff $\Pi = \emptyset$ since there are no infinite prefix descending chains. Then $(\Pi{\downarrow}){\uparrow} \subseteq \Pi{\uparrow}$ since $\Pi{\downarrow} \subseteq \Pi$ is minimal. For every $\pi \in \Pi$ there exists $\pi' \in \Pi{\downarrow}$ such that $\pi' \preceq \pi$ and by Lemma 2 $\pi{\uparrow} \subseteq \pi'{\uparrow}$ *i.e.* $\Pi{\uparrow} \subseteq (\Pi{\downarrow}){\uparrow}$. Therefore $\Pi{\uparrow} = (\Pi{\downarrow}){\uparrow}$. Consider Π' as in the enunciate, then, for every $\pi \in \Pi$ there exists $\pi' \in \Pi{\downarrow}$ such that $\pi' \preceq \pi$ and in particular if $\pi \in \Pi'$. Uniqueness follows straightforwardly. $\qquad\square$

The set $\Pi{\downarrow}$ is called *minimal support of* Π and intuitively correspond to the "minimal" set of finite executions needed to completely characterize the behavior captured by Π and the complete paths it induces. Any other path of Π is therefore redundant (*cf.* Example 3).

The idea of which complete paths are distinguishable and then "measurable" (*i.e.* that can be given weight) is captured precisely by the notion of σ-algebra. In fact, the set of all cones Γ (together with the emptyset) induce a σ-algebra, as they form a *semiring of sets* (in the sense of [44]).

8

Lemma 4. *The set $\Gamma \cup \{\emptyset\}$ is a semiring of sets and uniquely induces a σ-algebra over* CPaths(x).

Proof (Sketch). $\Gamma \cup \{\emptyset\}$ is closed under finite intersections since cones are always either disjoint or one the subset of the other. Set difference follows from the existence of minimal supports. $\square$

As discussed before, in general the weight of $\Pi \subseteq$ FPaths(x) cannot be defined as the sum of the weights of its elements, due to redundancies. However, what we are really interested in is the *unique* set of behaviors described by Π, *i.e.* the complete paths it subsumes. Therefore we first extend ρ to *minimal* Π, as follows:

$$\rho(\Pi) \triangleq \sum_{\pi \in \Pi} \rho(\pi) \quad \text{for } \Pi \text{ minimal.}$$

then, for all Π, we simply take

$$\rho(\Pi) \triangleq \rho(\Pi\!\downarrow).$$

Because Π can be countably infinite, semiring addition *has to* support countable additions over these sets (*cf.* Remark 1).

3.2 Well-behaved semirings

Definition 10. *Let the semiring $\mathfrak{W}$ be endowed with a preorder $\sqsubseteq$. We call the semiring well-behaved if, and only if, for any two Π_1 and Π_2 the following holds:*

$$\Pi_1 \subseteq \Pi_2 \Rightarrow \rho(\Pi_1) \sqsubseteq \rho(\Pi_2).$$

If the semiring is well-behaved then addition unit 0 is necessarily the bottom of the preorder because $\rho(\emptyset) \triangleq 0$. Moreover, the semiring operations have to respect the preorder *e.g.*:

$$a \sqsubseteq b \Rightarrow a + c \sqsubseteq b + c.$$

As a direct consequence, annihilation of parallel is avoided by the *zerosumfree* property of the semiring *i.e.* the sum of weighs of proper transition always yield the weight of a proper transition where proper means different from the addition unit.

Well-behaved semirings are precisely *positively (partially) ordered semirings* and it is well known that these admit the natural preorder:

$$a \trianglelefteq b \overset{\triangle}{\iff} \exists c. a + c = b$$

which is respected by the semiring operation and has 0 as bottom. The natural preorder is the weaker preorder rendering a semiring positively ordered (hence well-behaved) where weaked means that for any such preorder $\sqsubseteq$ and elements a, b

$$a \trianglelefteq b \implies a \sqsubseteq b.$$

The converse holds only when also the other order is natural.

Lemma 5. *The natural preorder is the weaker preorder rendering the semiring well-behaved.*

Note that any idempotent semiring bares a natural preorder and hence is well-behaved and the same holds for every semiring considered in the examples illustrated in this paper (*cf.* Section 4). For instance, some arithmetic semirings like $(\mathbb{R}, +, 0, \cdot, 1)$ are not positively ordered because of negatives; moreover their are not ω-semirings (there is no limit for $1 + (-1) + 1 + (-1) \ldots$).

3.3 Weak $\mathfrak{W}$-bisimulation

Weak bisimulations weakens the notion of strong bisimulation by allowing sequences of silent action before and after any observable one. Then, we are now dealing with (suitable) paths instead of single transitions and the states are compared on the bases of how opportune classes of states are reached from these by means of the paths allowed (*i.e.* making some silent actions, before and after an observable, if any). Therefore, the notion of how a class state is reached and what paths can be used in doing this is crucial in the definition of the notion of weak bisimulation.

For instance, for non-deterministic LTSs, the question of how and if a class is reached coincides and then it suffices to find a (suitable) path leading to the class. This allows weak bisimulation for non-deterministic LTSs to rely on the reflexive and transitive closure of τ-labelled transition of a system (*cf.* Definition 12) to blur the distinction between sequences of silent actions which can then be "skipped". In fact, the τ-closure at the base of (3) defines a new LTS over the same state space of the previous and such that every weak bisimulation for this new system is a weak bisimulation for the given one and vice versa.

In [11] Buchholz and Kemper extends this notion to a class of automatons weighted over suitable semirings *i.e.* those having operations commutative and idempotent (*e.g.* $w + w = w$). This class includes interesting examples such as the boolean and bottleneck semiring (*cf.* Section 4.4) but not the semiring of non-negative real numbers and therefore does not cover the cases of fully probabilistic systems. Modulo some technicality connected to initial and accepting states, their results can be extended to labelled transition systems and holds also for LTSs weighted over suitable semirings.

Their interesting construction relies on the τ-closure of a system and it is known that this closure does not cover the general case. For instance, it can not be applied to recover weak bisimulation for generative systems as demonstrated by Baier and Hermanns (*cf.* [2]). The following example gives an intuition of the issue.

Example 3. Consider the $\mathfrak{W}$-LTS below.
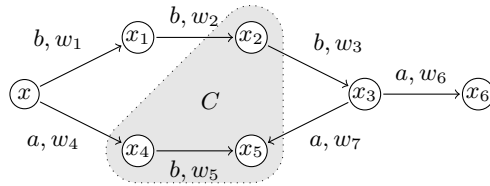
There are four finite paths going from state the x to the class C. Their weights are:

$$\rho(x \xrightarrow{b} x_1 \xrightarrow{b} x_2) = w_1 \cdot w_2$$

$$\rho(x \xrightarrow{b} x_1 \xrightarrow{b} x_2 \xrightarrow{b} x_3 \xrightarrow{a} x_5) = w_1 \cdot w_2 \cdot w_3 \cdot w_7$$

$$\rho(x \xrightarrow{a} x_4) = w_4$$

$$\rho(x \xrightarrow{a} x_4 \xrightarrow{b} x_5) = w_4 \cdot w_5$$

Let us suppose to define the weight of the set of these paths as the sum of its elements weights and suppose that the system is generative; then the probability of reaching C from x would exceed 1. Likewise, in the case of a stochastic system, the rate of reaching C cannot consider paths passing through C before ending in it. If we are interested in how C is reached from x with actions yielding a trace in the set b^*ab^*, paths $w_1 \cdot w_2$ and $w_5 \cdot w_6$ are ruled out because the first has a different trace and the second reaches C before it ends.

Then, given a set of traces T, a state x and a class of states C, the set of finite paths of the given transition system reaching C from x with trace in T that should be considered is:

$$\langle x, T, C \rangle \triangleq \left\{ \pi \,\middle|\, \begin{array}{l} \pi \in \mathrm{FPaths}(x),\ \mathrm{last}(\pi) \in C,\ \mathrm{trace}(\pi) \in T, \\ \forall \pi' \preceq \pi : \mathrm{trace}(\pi') \in T \Rightarrow \mathrm{last}(\pi') \notin C \end{array} \right\}$$

since these are all and only the finite executions of the system starting going from x to C with trace in T and never passing through C except for their last state. Redundancies highlighted in the example above are ruled out since no execution path in this set is the prefix of an other in the same set. In particular $\wr x, T, C \wr$ is the minimal support of the set of all finite paths reaching C from x with trace in T:

$$\wr x, T, C \wr = \{\pi \mid \pi \mathrm{FPaths}(x), \mathrm{last}(\pi) \in C, \mathrm{trace}(\pi) \in T\} \downarrow.$$

Therefore, weight functions can be consistently extended to these sets by point-wise sums:

$$\rho(\wr x, T, C \wr) = \sum_{\pi \in \wr x, T, C \wr} \rho(\pi).$$

The sum is at most countable since $\mathrm{FPaths}(x)$ so is $\mathrm{FPaths}(x)$ and $\wr x, T, C \wr \subseteq \mathrm{FPaths}(x)$. Then, the addition operation of the semiring will support countable sums as discussed in Remark 1.

When clear from the context, we may omit the bag brackets from $\rho(\wr x, T, C \wr)$.

We are now ready to state the notion of weak bisimulation of a labelled transition system weighted over any semiring admitting sums over (not necessarily every) countable family of weights. The notion we propose relies on the weights of paths reaching every class in the relation but making at most one observable and hence the importance of defining sets of paths reaching a class consistently.

Definition 11 (Weak $\mathfrak{W}$-bisimulation). *Let $(X, A+\{\tau\}, \rho)$ be a LTS weighted over the semiring $\mathfrak{W}$. A weak $\mathfrak{W}$-bisimulation is an equivalence relation R on X such that for all $x, x' \in X$, $(x, x') \in R$ implies that for each label $a \in A$ and each equivalence class C of R:*

$$\rho(x, \tau^* a \tau^*, C) = \rho(x, \tau^* a \tau^*, C)$$
$$\rho(x, \tau^*, C) = \rho(x, \tau^*, C).$$

States x and x' are said to be weak $\mathfrak{W}$-bisimilar (or just weak bisimilar), written $x \approx_{\mathfrak{W}} x'$, if there exists a weak $\mathfrak{W}$-bisimulation R such that xRx'.

The approach we propose applies to other behavioural equivalences. For instance, delay bisimulation can be recovered for WLTSs by simply considering in the above definition of weak bisimulations sets of paths of the sort of $\wr x, \tau^*, C \wr$ and $\wr x, \tau^* a, C \wr$. The notion of branching bisimulation relies on paths with the same traces of those considered for defining weak bisimulation but with some additional constraint on the intermediate states. In particular, the states right before the observable a have to be in the same equivalence class and likewise the states right after it. Definition 11 is readily adapted to branching bisimulation by considering these particular subsets of $\wr x, \tau^* a \tau^*, C \wr$.

4 Examples of weak $\mathfrak{W}$ bisimulation

In this Section we instantiate Definition 11 to the systems introduced in Section 2 as instances of LTSs weighted over commutative ω-monoids.

4.1 Non-deterministic systems

Let us recall the usual definition of weak bisimulation for LTS [32].

Definition 12 (Weak non-deterministic bisimulation). *An equivalence relation $R \subseteq X \times X$ is a weak (non-deterministic) bisimulation on $(X, A + \{\tau\}, \rightarrow)$ iff for each $(x, x') \in R$, label $\alpha \in A + \{\tau\}$ and equivalence class $C \in X/R$:*

$$\exists y \in C.x \stackrel{\alpha}{\Rightarrow} y \iff \exists y' \in C.x' \stackrel{\alpha}{\Rightarrow} y' \tag{3}$$

where $\Rightarrow \subseteq X \times (A \uplus \{\tau\}) \times X$ is the well-known τ-reflexive and τ-transitive closure of the transition relation $\rightarrow$. Two states x and x' are said weak bisimilar iff there exists a weak non-deterministic bisimulation relation $\approx_n$ such that $x \approx_n y$.

Clearly, a weak bisimulation is a relation on states induced by a strong bisimulation of a suitable LTS with the same states and actions. In particular, weak bisimulations for $(X, A+\{\tau\}, \rightarrow)$ are strong bisimulations for $(X, A+\{\tau\}, \Rightarrow)$ and viceversa. The transition system $(X, A+\{\tau\}, \Rightarrow)$ is sometimes referred as saturated or weak (*e.g.* in [22]). This observation is at the base of some algorithmic and coalgebraic approaches to weak non-deterministic bisimulations (*cf.* Section 5 and Section 6 respectively).

Section 2.1 illustrated that non-deterministic LTSs are 2-WLTSs. The commutative monoid 2 is part of the *boolean semiring* of logical values under disjunction and conjunction $(\{\mathtt{tt}, \mathtt{ff}\}, \vee, \mathtt{ff}, \wedge, \mathtt{tt})$ which we shall also denote as 2. Then, by straightforward application of the definitions, the notions of weak non-deterministic bisimulation and weak 2-bisimulation coincide.

Proposition 6. *Definition 12 is equivalent to Definition 11 with $\mathfrak{W} = 2$.*

It easy to check that a similar correspondence holds for branching and delay bisimulations.

4.2 Probabilistic systems

In the definition of weak bisimulation for fully probabilistic systems we are interested in the probability of reach a class of states. This aspect is present also in the case of strong bisimulation, but things become more complex for weak equivalences due to silent actions and multi-step executions. Moreover, σ-additivity is no longer available since the probability of reaching a class of states is not the sum of the probabilities of reaching every single state in that class. (On the contrary, a class is reachable if any of its state is so which is the property we are interested in when dealing with non-deterministic systems.)

Weak bisimulation for fully probabilistic systems was introduced by Baier and Hermanns in [4, 2]. Here we recall briefly their definition; we refer the reader to *loc. cit.* for a detailed presentation.

Definition 13 (Weak probabilistic bisimilarity [4, 2])**.** *Given a fully probabilistic system $(X, A+\{\tau\}, \mathrm{P})$, an equivalence relation R on X is a* weak (probabilistic) bisimulation *iff for $(x, x') \in R$, for any $a \in A$ and any equivalence class $C \in X/R$:*

$$\mathrm{Prob}(x, \tau^* a \tau^*, C) = \mathrm{Prob}(x', \tau^* a \tau^*, C)$$
$$\mathrm{Prob}(x, \tau^*, C) = \mathrm{Prob}(x', \tau^*, C).$$

Two states x and x' are said weak bisimilar *iff there exists a weak probabilistic bisimulation relation $\approx_p$ such that $x \approx_p y$.*

The function Prob is the extension over finite execution paths of the unique probability measure induced by P over the σ-field of the basic cylinders of complete paths.

Proposition 7. *Definition 12 is equivalent to Definition 11 with $\mathfrak{W} = (\mathbb{R}_0^+, +, 0, \cdot, 1)$.*

The function P is a weight function such that $\mathrm{P}(x, \_, \_)$ is a probability measure (or the constantly 0 measure) which extends to the unique σ-algebra on $\mathrm{CPaths}(x)$ (Lemma 4). This defines precisely *Prob*. In particular, for any $x \in X$ and $\Pi \subseteq \mathrm{FPaths}(x)$ $Prob(\Pi) = Prob(\Pi\downarrow) = \mathrm{P}(\Pi\downarrow) = \mathrm{P}(\Pi)$ where P is seen as the weigh function of a $\mathbb{R}_0^+$-LTS.

4.3 Stochastic systems

As we have seen in Section 2.2, stochastic transition systems can be captured as WLTSs over $(\overline{\mathbb{R}}_0^+, +, 0)$ by describing the exponential time distributions of a CTMC by their rates [26]. Unfortunately, this does not extend to paths because the sequential composition of two exponential distributions does not yield an exponential distribution, and hence it can not be represented by an element of $\overline{\mathbb{R}}_0^+$. Moreover, there are stochastic systems (*e.g.* TIPP [15], SPADES [12]) whose transition times follow generic probability distributions.

To overcome this shortcoming, in this Section we introduce a semiring of weights called *stochastic variables* which allows to express stochastic transition system with generic distributions as WLTSs. Then the results of this theory can be readily applied to define various behavioural equivalences, ranging from strong bisimulation to trace equivalence, for all these kind of systems. In particular, we define *weak stochastic bisimulation* by instantiating Definition 11 on the semiring of stochastic variables.

The carrier of the semiring structure we are defining is the set $\mathbb{T}$ of *transition-time random variables i.e.* random variables on the nonnegative real numbers (closed with infinity) which describes the nonnegative part of the line of time.

Given two (possibly dependent) random variables X and Y from $\mathbb{T}$, let $\min(X, Y)$ be the minimum random variable yielding the minimum between X and Y. If the variables X and Y characterize the time required by two transitions then their combined effect is defined by the stochastic race between the two transitions; a race that is "won" by the transition completed earlier and hence the minimum. For instance, given two stochastic transitions $x \xrightarrow{X} x'$ and $y \xrightarrow{Y} y'$ the transition time for their "combination" going from $\{x, y\}$ to $\{x', y'\}$ is characterized by the random variable $\min(X, Y)$ *i.e.* the overall time is given by the first transition to be completed on the specific run.

Minimum random variables defines the operation min over $\mathbb{T}$ with a constantly $+\infty$ continuous random variable $\mathcal{T}_{+\infty}$ (its density is the Dirac delta function $\delta_{+\infty}$) as the unit. Random variables of the sort of $\mathcal{T}_{+\infty}$ are self-independent and since they always always yield $+\infty$ we shall make no distinction between them and refer to *the* $\mathcal{T}_{+\infty}$ random variable. In general, time-transition variables do not have to be self-independent since the events they describe usually depends on themselves. Intuitively, it is like racing against ourself *i.e.* we are the only racer and therefore $\min(X, X) = X$. Formally:

$$\mathbb{P}(\min(X, X) > t) = \mathbb{P}(X > t \cap X > t) = \mathbb{P}(X > t) \cdot \mathbb{P}(X > t \mid X > t) = \mathbb{P}(X > t).$$

Let X and Y be two continuous random variables from $\mathbb{T}$ with probability density functions f_X and f_Y respectively. The density $f_{\min(X,Y)}$ describing $\min(X, Y)$ is:

$$f_{\min(X,Y)}(z) = f_X(z) + f_Y(z) - f_{X,Y}(z, z).$$

When X and Y are independent (but not necessarily *i.i.d.*) $f_{\min(X,Y)}$ can be simplified as:

$$f_{\min(X,Y)}(z) = f_X(z) \cdot \int_z^{+\infty} f_Y(y)\mathrm{d}y + f_Y(z) \cdot \int_z^{+\infty} f_X(x)\mathrm{d}x.$$

Intuitively, the likelihood that one variable is the minimum must be "weighted" by the probability that the other one is not. In particular, for independent exponentially distributed variables X and Y, $\min(X, Y)$ is exponentially distributed and its rate is the sum of the rates of the negative exponentials characterizing X and Y. Therefore, the commutative monoid $(\mathbb{T}, \min, \mathcal{T}_{+\infty})$ faithfully generalizes the monoid $(\overline{\mathbb{R}}_0^+, +, 0)$ used in Section 2.2 to capture CTMCs as WLTSs

During the execution of a given path, the time of every transition in the sequence sums to the overall time. Therefore, the transition time for *e.g.* $x \xrightarrow{X} y \xrightarrow{Y} z$ is characterized by the random variable $X + Y$ sum of the variable characterizing the single transitions composing the path. Sum and the constantly 0 continuous variable $\mathcal{T}_0$ define a commutative monoid over $\mathbb{T}$. The operation has to be commutative because the order a path imposes to its steps does not change the total time of execution.

Let X and Y be two continuous random variables from $\mathbb{T}$ with probability density functions f_X and f_Y respectively. The probability density function f_{X+Y} is:

$$f_{X+Y}(t) = \int_0^t f_{X,Y}(s, t - s)\mathrm{d}s$$

and, if X and Y are independent (but not necessarily *i.i.d.*), f_{X+Y} is the convolution:

$$f_{X+Y}(t) = \int_0^t f_X(s) \cdot f_Y(t - s)\mathrm{d}s.$$

It is easy to check that sum distributes over minimum:

$$X + \min(Y, Z) = \min(X + Y, X + Z)$$

by taking advantage of the latter operation being idempotent. Then, because of sum being commutative, left distributivity implies right one (and vice versa). Thus $\mathfrak{S} \triangleq (\mathbb{T}, \min, \mathcal{T}_{+\infty}, +, \mathcal{T}_0)$ is a (commutative and idempotent) semiring and stochastic systems can be read as $\mathfrak{S}$-LTS. This induces immediately a strong bisimulation (by instantiating Definition 2) which corresponds to strong stochastic bisimulations on rated LTS (Definition 6). Moreover, following Definition 11, we can readily define the weak stochastic bisimulation as the weak $\mathfrak{S}$-bisimulation.

In literature there are some (specific and ad hoc) notions of weak bisimilarity for stochastic systems. The closest to our is the one given by Bernardo et al. for CTMCs extended with passive rates and *instantaneous* actions [6, 5]. Their definition is finer than our weak $\mathfrak{S}$-bisimulation since they allow to merge silent actions only when these are instantaneous and hence unobservable also *w.r.t.* the time. Instead, in our definition sequences of τ actions are equivalent as long as their overall "rates" are the same (note that in general, the convolution of exponentially distributed random variables is no longer exponentially distributed but an hyper-exponential). In [5], Bernardo et al. relaxed the definition given in [6] to account also for non-instantaneous τ-transitions. However, to retain exponentially distributed variables, they approximate hyper-exponentials with exponentials with the same average. This approach allows them to obtain a saturated system that still is a CTMC but loosing precision since, in general, the average is the only momentum preserved during the operation. On the opposite, our approach does not introduce any approximation.

In [30] López and Núñez proposed a definition of weak bisimulation for stochastic transition systems with generic distributions. Their (rather involved) definition is a refinement of the notion they previously proposed in [31] and relies on the reflexive and transitive closure of silent transitions. However, their definition of strong bisimulation does not correspond to the results from the theory of WLTSs, so neither the weak one does.

4.4 Other examples

The definition of weak $\mathfrak{W}$-bisimulation applies to many other situations. In the following we briefly illustrate some interesting cases.

Tropical and arctic semirings These semirings are used very often in optimization problems, especially for task scheduling and routing problems. Some examples are: $(\overline{\mathbb{R}}, \min, +\infty, +, 0)$; $(\overline{\mathbb{R}}, \max, -\infty, +, 0)$; $(\overline{\mathbb{R}}, \min, +\infty, \max, -\infty)$.

In these contexts, weak bisimulation would allow to abstract from "unobservable" tasks *e.g.* internal tasks and treat a cluster of machines as a single one, reducing the complexity of the problem.

Truncation semiring $(\{0, \ldots, k\}, \max, 0, \min\{\_ + \_, k\}, k)$. It is variant of the above ones, and it is used to reason "up-to" a threshold k. A weak bisimulation for this semiring allows us to abstract from how the threshold is violated, but only if this happens.

Probabilistic semiring Another semiring used for reasoning about probabilistic events is $([0, 1], \max, 0, \cdot, 1)$. This is used to model the maximum likelihood of events, *e.g.* for troubleshooting, diagnosis, failure forecasts, worse cases, etc. A weak bisimulation on this semiring allows to abstract from "unlikely" events, focusing on the most likely ones.

Formal languages A well-known semiring is that of formal languages over a given alphabet $(\wp(\Sigma^*), \cup, \emptyset, \circ, \varepsilon)$. Here, a weak bisimulation is a kind of *determinization w.r.t.* to words assigned to τ transitions.

5   A parametric algorithm for computing weak $\mathfrak{W}$-bisimulations

In this section we present an algorithm for computing weak $\mathfrak{W}$-bisimulation equivalence classes which is parametric in the semiring structure $\mathfrak{W}$. Being parametrized, the same algorithm can be used in the mechanized verification and analysis of many kinds of systems. This kind of algorithms is often called *universal* since they do not depend on any particular numerical domain nor its machine representation. In particular, algorithms parametric over a semiring structure have been successfully applied to other problems of computer science, especially in the field of system analysis and optimization (*cf.* [29]).

The algorithm we present is a variation of the well-known Kanellakis-Smolka's algorithm for deciding strong non-deterministic bisimulation [23]. Our solution is based on the same refinement technique used for the *coarsest stable partition*, but instead of "strong" transitions in the original system we consider "weakened" or saturated ones. The idea of deciding weak bisimulation by computing the strong bisimulation equivalence classes for the saturated version of the system has been previously and successfully used *e.g.* for non-deterministic or probabilistic weak bisimulations [2]. The resulting complexity is basically that of the coarsest stable partition problem plus that introduced by the construction of the saturated transitions. The last factor depends on the properties and kind of the system and, in our case, on the properties of the semiring $\mathfrak{W}$ (the algorithm and its complexity will be discussed with more detail in Section 5.2).

Before outlining the general idea of the algorithm let us introduce some notation. For a finite set X we denote by $\mathcal{X}$ a partition of it *i.e.* a set of pairwise disjoint sets $B_0, \ldots, B_n$ covering X:

$$X = \biguplus \mathcal{X} = \biguplus \{B_0, \ldots, B_n\}.$$

We shall refer to the elements of the partition $\mathcal{X}$ as *blocks* or *classes* since every partition induces an equivalence relation $\bigcup_{B \in \mathcal{X}} B \times B$ on X and vice versa.

Given a finite $\mathfrak{W}$-LTS $(X, A + \{\tau\}, \rho)$ the general idea for deciding weak $\mathfrak{W}$-bisimulation by partition refinement is to start with a partition of the states $\mathcal{X}_0$ coarser than the weak bisimilarity relation *e.g.* $\{X\}$ and then successively refine the partition with the help of a *splitter* (*i.e.* a witness that the partition is not stable *w.r.t.* the transitions). This process eventually yields a partition $\mathcal{X}_k$ being the set of equivalence classes of the weak bisimilarity. A splitter of a partition $\mathcal{X}$ is a pair made of an action and a class of $\mathcal{X}$ that violates the condition for $\mathcal{X}$ to be a weak bisimulation. Reworded, a pair $\langle \alpha, C \rangle \in (A + \{\tau\}) \times \mathcal{X}$ is a splitter for $\mathcal{X}$ if, and only if, there exist $B \in \mathcal{X}$ and $x, y \in B$ such that:

$$\rho(x, \hat{\alpha}, C) \neq \rho(y, \hat{\alpha}, C) \tag{4}$$

where $\hat{\alpha}$ is a short hand for the sets of traces τ^* and $\tau^* a \tau^*$ when $\alpha = \tau$ and $\alpha = a \in A$ respectively. Then $\mathcal{X}_{i+1}$ is obtained from $\mathcal{X}_i$ splitting every[3] $B \in \mathcal{X}_i$ accordingly to the selected splitter $\langle \alpha, C \rangle$.

$$\mathcal{X}_{i+1} \triangleq \bigcup \left\{ B / \underset{\alpha, C}{\approx} \mid B \in \mathcal{X}_i \right\} \tag{5}$$

where $\underset{\alpha, C}{\approx}$ is the equivalence relation on states induced by the splitter and such that:

$$x \underset{\alpha, C}{\approx} y \overset{\triangle}{\iff} \rho(x, \hat{\alpha}, C) = \rho(y, \hat{\alpha}, C).$$

Note that the block B can be split in more than two parts (which is the case of non-deterministic systems) since splitting depends on weights of outgoing weak transitions.

---

[3] In Kanellakis and Smolka's algorithm, only the block B is split but in our case we need to evaluate every block anyway because of saturation, *cf.* Section 5.1.

5.1 Computing weak transitions

The algorithm outlined above follows the classical approach to the coarsest stable partition problem where stability is given in terms of weak weighted transitions like $\langle x, \tau^*, C \rangle$ (and in general weighted sets of paths *e.g.* $\langle x, T, C \rangle$) but nothing is assumed on how these values are computed. In this section, we show how weights of weak transitions can be obtained as solutions of systems of linear equations over the semiring $\mathfrak{W}$. Clearly, for some specific cases and sets of paths, there may be more efficient *ad-hoc* technique (*e.g.* saturated transitions can be precomputed for non-deterministic LTSs) however the linear system at the core of our algorithm is a general and flexible solution which can be readily adapted to other observational equivalences (*cf.* Example 4).

Let C be a class. For every $x \in X$ and $\alpha \in A + \{\tau\}$ let x_α be a variable with domain the semiring carrier. Intuitively, once solved, these will represent:

$$x_\tau = \rho(\langle x, \tau^*, C \rangle) \qquad x_a = \rho(\langle x, \tau^* a \tau^*, C \rangle)$$

The linear system is given by the equation families (6), (7) and (8) which capture exactly the finite paths yielding the cones covering weak transitions.

$$x_\tau = 1 \qquad\qquad\qquad\qquad\qquad \text{for } x \in C \qquad\qquad (6)$$

$$x_\tau = \sum_{y \in X} \rho(x \xrightarrow{\tau} y) \cdot y_\tau \qquad\qquad \text{for } x \notin C \qquad\qquad (7)$$

$$x_a = \sum_{y \in X} \rho(x \xrightarrow{a} y) \cdot y_\tau + \sum_{y \in X} \rho(x \xrightarrow{\tau} y) \cdot y_a \qquad\qquad (8)$$

The system is given as a whole but it can be split in smaller sub-systems improving the efficiency of the resolution process. In fact, unknowns like x_a depend only on those indexed by τ or a and unknowns like x_τ depend only on those indexed by τ. Hence instead of a system of $|A + \{\tau\}| \cdot |X|$ equations and unknowns, we obtain $|A + \{\tau\}|$ systems of $|X|$ equations and unknowns by first solving the sub-system for x_τ and then a separate sub-system of each action $a \in A$ (where x_τ are now constant).

Example 4 (Delay bisimulation). *Delay bisimulation is defined at the general level of WLTSs simply by replacing $\langle x, \tau^* a \tau^*, C \rangle$ with $\langle x, \tau^* a, C \rangle$ in Definition 11. Then, delay bisimulation equivalence classes can be computed with the same algorithm simply by changing the saturation part at its core. Weights of sets like $\langle x, \tau^* a, C \rangle$ are computed as the solution to the linear equation system:*

$$x_a = \sum_{y \in X} \rho(x \xrightarrow{\tau} y) \cdot y_a + \sum_{y \in C} \rho(x \xrightarrow{a} y).$$

5.1.1 Solvability

Decidability of the algorithm depends on the solvability the equation system at its core. In particular, on the existence and uniqueness of the solution. In section we prove that this holds for every positively ordered ω-semiring. The results can be extended to σ-semirings provided that their σ-algebra covers the countable families used by Theorem 10.

The linear equation systems under consideration bare a special form: they have exactly the same number of equations and unknowns (say n) and every unknown appears alone on the left side of exactly one equation. Therefore, these systems are defining an operator

$$F(x) = M \times x + b \qquad\qquad (9)$$

over the space of n-dimensional vectors W^n where M and b are a n-dimensional matrix and vector respectively defined by the equations of the system. Then, the solutions of the system are precisely the fix-points of the operator F and since the number of equations and unknowns is the same, if F has a fix-point, it is unique.

Let the semiring $\mathfrak{W}$ be positively ordered. These semirings admit a natural preorder $\trianglelefteq$ which subsumes any preorder $\sqsubseteq$ respecting the structure of the semiring; hence we restrict ourselves to the former. The point-wise extension of $\trianglelefteq$ to n-dimensional vectors defines the partial order with bottom $(W^n, \dot{\trianglelefteq}, 0^n)$; suprema are lifted pointwise from $(W, \trianglelefteq, 0)$ where are sum-defined. Therefore, ω-chains suprema exists only under the assumption of addition over at most countable families and viceversa.

Lemma 8. $(W^n, \dot{\trianglelefteq}, 0^n)$ *is ω-complete iff $\mathfrak{W}$ admits countable sums.*

The operator F manipulates its arguments only by additions and constant multiplications which respect the natural order. Thus F is monotone with respect to $\dot{\trianglelefteq}$. Moreover, F preserves ω-chains suprema (and in general ω-families) because suprema for $\trianglelefteq$ are defined by means of additions and the order is lifted point-wise.

Lemma 9. *The operator F over $(W^n, \dot{\trianglelefteq}, 0^n)$ is Scott-continuous.*

Finally, we can state the main result of this Section from which decidability follows as a corollary.

Theorem 10. *Systems in the form of* (9) *have unique solutions if the underlying semiring is well-behaved and ω-complete.*

Proof. By Lemma 8, Lemma 9 and Kleene Fix-point Theorem F has a least fix point. Because the linear equation system has the same number of equations and unknowns, this solution is unique. $\qquad\qquad\square$

The linear equation systems defined by the equation families (6), (7) and (8) have exactly one solution and hence the algorithm proposed is decidable. Moreover this holds also for any behavioural equivalence whose saturation can be expressed in a similar way *e.g.* delay bisimulation (*cf.* Example 4).

5.1.2 Adequacy

If $x \in C$, then the empty execution ε is the only element of the set $\langle x, \tau^*, C \rangle$, (by definition of reachability) $\rho(\varepsilon)$ is the value of the 0-fold multiplication *i.e.* the unit 1. This case falls under (6) and hence x_τ is $\rho(x, \tau^*, C)$ when $x \in C$.

On the other hand, if $x \notin C$, then every path reaching C from x needs to have length strictly greater than 0; reworded, it starts with a transition $x \xrightarrow{\tau} y$ and from y heads towards C. The weight of $\langle x, \tau\tau^*, C \rangle$ is the sum of the weights of its paths which are themselves the ordered multiplication of their steps. Then by grouping paths by their second state the remaining parts are exactly the paths in the set $\langle y, \tau^*, C \rangle$. Then we obtain the unfolding

$$\rho(\langle x, \tau\tau^*, C \rangle) = \sum_{y \in X} \rho(x \xrightarrow{\tau} y) \cdot \rho(\langle y, \tau^*, C \rangle)$$

which recursively defines the weight of these sets as the unfolding of executions. In particular, the base case is precisely (7) and the inductive one is (6).

Every path in the set $\langle x, \tau^* a \tau^*, C \rangle$ contains exactly one transition labelled by the action a and hence it has a transition, to some state y, and is labelled with either a or τ. In the first case , the observable a is consumed and remaining path is necessarily in the set $\langle y, \tau^*, C \rangle$ covered above. In the second case, the only observable of the path has not been consumed yet and thus the remaining part of the path should be in the set $\langle y, \tau^* a \tau^*, C \rangle$ completing the case for (8).

Proposition 11. *Let $\mathfrak{W}$ be a positively ordered ω-semiring. For any C, α and x, solutions for* (6), (7) *and* (8) *are exactly the weights of $\langle x, \hat{\alpha}, C \rangle$.*

```
 1: 𝒳 ← {X}
 2: 𝒳′ ← ∅
 3: repeat
 4:    changed ← false
 5:    𝒳″ ← 𝒳
 6:    for all C ∈ 𝒳 \ 𝒳′ do
 7:       for all α ∈ A + {τ} do
 8:          if ⟨α, C⟩ is a split then
 9:             𝒳 ← ⋃{B/≈ | B ∈ 𝒳}
                        α,C
10:             changed ← true
11:          end if
12:       end for
13:    end for
14:    𝒳′ ← 𝒳″
15: until not changed
16: return 𝒳
```

Figure 1: The algorithm for weak 𝔚-bisimulation.

5.2   The algorithm and its complexity

In this section we describe the algorithm and study its worst case complexity. The algorithm and the resulting analysis follow the structure of the Kanellakis-Smolka's result. However, some assumptions available in the case of strong bisimulation for non-deterministic systems are not available in this settings. For instance, transitions have to be computed on the fly. Moreover, like many other algorithms parametrized over semirings, no hypotheses are made over the numerical domain nor over its machine representation. As a consequence, we can not assume constant-time random access data-structures or linearly order the elements of the semiring. However, since many practical semirings admit total-orderings and efficient data structures, we will describe also this second case providing a more efficient version of the algorithm for the general case.

The first algorithm we propose is reported in Figure 1. Given a finite 𝔚-LTS $(X, A + \{\tau\}, \rho)$ as input, it returns a partition $\mathcal{X}$ of X inducing a weak 𝔚-bisimulation for the system.

The partition $\mathcal{X}$ is initially assumed to have the set of states X as its only block and corresponds to the assumption of the largest possible equivalence relation on X being also a weak bisimulation. In general, any partition coarser than some weak bisimulation would be a suitable initial partition.

The purpose of the two auxiliary partitions $\mathcal{X}'$ and $\mathcal{X}''$ is to keep track of which classes were added to $\mathcal{X}$ during the previous iteration of the repeat-until loop and thus avoiding to reuse a split candidate. We used these additional partitions for readability but the same result may be achieved, for instance, having two colours distinguishing blocks already checked. Moreover, $\mathcal{X}'$ and $\mathcal{X}''$ make the flag *changed* redundant.

The algorithms iterates over each split candidate $\langle \alpha, C \rangle$ and tries to split the partition by checking whatever (4) holds. If the partition "survives" to every split test then it is stable and in particular it describes a weak 𝔚-bisimulation relation. The saturated transitions required to test $\langle \alpha, C \rangle$ are computed by solving the linear equations system described before. Overall, we have to solve $|A| + 1$ systems of $|X|$ linear equations and unknowns for each C.

The complexity of solving these systems depends on the underling semiring structure. For instance, solving a system over the semiring of non-negative real numbers is in P [1], whereas solving a system over the tropical (resp. arctic) semiring is in NP ∩ coNP (*cf.* [16]). Since the algorithm is parametrized by the semiring, its complexity will be parametrized by the one introduced by the solution of these linear equation systems. Therefore we shall denote by $\mathcal{L}_{\mathfrak{W}}(n)$ the complexity of solving a system of n linear equations in n variables over 𝔚.

Remark 5. *The complexity of the split test can be made preciser since we are not solving a general linear system, but a specific sub-class of these. For instance, solving a linear system*

```
 1: 𝒳 ← {X}
 2: 𝒳' ← ∅
 3: repeat
 4:    changed ← false
 5:    for all C ∈ 𝒳 \ 𝒳' do
 6:       for all α ∈ A + {τ} do
 7:          compute and sort ρ(x, α̂, C) by block and weight
 8:       end for
 9:       if there is any split then
10:          𝒳' ← 𝒳
11:          𝒳 ← refine(𝒳, C)
12:          changed ← true
13:       end if
14:    end for
15: until not changed
16: return 𝒳
```

Figure 2: An alternative algorithm for linearly ordered blocks and weights.

over the boolean semiring is NP-Complete in general, whereas we are interested in a specific subclass of those encoding a reachability problem over a directed graph which is in P.

Let n and m denote the cardinality of states and labels respectively. For each block C used to generate splits, there are exactly m candidates requiring to solve m split tests and perform at most m updates to $\mathcal{X}$. Splits can be thought describing a tree whose nodes are the various blocks encountered by the algorithm during its execution and whose leaves are exactly the elements of the final partition. Because the cardinality of $\mathcal{X}$ is bound by n, the algorithm can encounter at most $\mathcal{O}(n)$ blocks during its entire execution and hence it performs at most $\mathcal{O}(n)$ updates of $\mathcal{X}$ (which happens when splits describe a perfect tree with n leaves). Therefore, in the worst case, the algorithm does $\mathcal{O}(nm)$ split tests and $\mathcal{O}(n)$ partition refinements. Partition refinements and checks of (4) can be both done in $\mathcal{O}(n^2)$ without any additional assumption about X, A and $\mathfrak{W}$ nor the use of particular data structures or primitives. Therefore the asymptotic upper bound for time complexity of the proposed algorithm is $\mathcal{O}(nm(\mathcal{L}_{\mathfrak{W}}(n) + n^2))$ where $\mathcal{L}_{\mathfrak{W}}(n)$ is the upper bound for the complexity introduced by computing the weak transitions for a given set of states.

The time complexity can be lowered by means of more efficient representations of systems, partitions and weights. For instance, the structure of every semiring can be used to define an ordering for its elements (*cf.* [17]) allowing the use of lookup data structures. Under the assumption of some linear ordering for weights and blocks (at least within the same partition) the operations of refinement and split testing can be carried out more efficiently by sorting lexicographically the transitions ending in the splitting block C. The resulting algorithm is reported in Figure 2.

This allows the algorithm to carry out the refinement of $\mathcal{X}$ while it is reading the lexicographically ordered list of the saturated transitions. In fact, a block B is split by $\langle \alpha, C \rangle$ if the list contains different weights in the portion of the list where B appears. A change in the weights correspond to two states x and y such that (4) holds. For each $\langle \alpha, C \rangle$ there are at most n weak transitions $\rho(x, \hat{\alpha}, C)$ and these are sorted in $\mathcal{O}(n\ln(n))$ – or in $\mathcal{O}(n)$ using a classical algorithm from [1]. On the worst case the algorithm encounters $\mathcal{O}(n)$ blocks during its entire execution yielding a worst case time complexity in $\mathcal{O}(nm(\mathcal{L}_{\mathfrak{W}}(n) + n))$.

Overall, we have proved the following result:

Proposition 12. *The asymptotic upper bound for time complexity of the algorithm is in $\mathcal{O}(nm(\mathcal{L}_{\mathfrak{W}}(n) + n^2))$, for the general case, and in $\mathcal{O}(nm(\mathcal{L}_{\mathfrak{W}}(n) + n))$ given a linear ordering for blocks and weights. Both algorithms have space complexity in $\mathcal{O}(mn^2)$.*

6 Coalgebraic perspective

In this Section we illustrate the categorical construction behind Definition 11. The presentation is succinct due to space constraints but it is based on general results from coalgebraic theory. In particular, we define weak bisimulations as cocongruences of saturated or weak systems extending the elegant approach proposed by Silva and Westerbaan in [36].

This is not the first work on a coalgebraic perspective of weak bisimulations coalgebraically, as in the recent years there have been several works in this direction. In general, the approach is to recover weak bisimulation as the coalgebraic bisimulation of saturated systems. In [38, 39] Sokolova et al. studied the case of action-based coalgebras and demonstrated their results on the cases of non-deterministic and fully-probabilistic systems. In particular, the latter required to change the category of coalgebras. Recently, Brengos [9, 10] proposed an interesting construction based on ordered-functors which yields saturated coalgebras for the same behavioural functor. Both these constructions are parametric in the notion of saturation and are therefore way more general; [10] describes an algebraic structure and some conditions yielding precisely saturations for weak bisimulations. However, this approach does not cover the case of generative and stochastic systems [9, Sec. 6] yet. In [37] Sobociński describes a neat account of weak (bi)simulation for non-deterministic systems and proves that saturation via the double-arrow construction (*i.e.* τ-closure) results from a suitable change of base functor having a left adjunct in the 2-categorical sense.

Likewise, we rely on saturation of the given systems but we do not require any additional parameter. Moreover, we base our definition on cocongruences which allow us to work explicitly with the equivalence classes and saturate the given coalgebras such that these describes how each class is reached by each state without the need to alter the behavioural functor.

Our saturation construction builds on the account of ϵ-transitions recently given in [36] and on the neat coalgebraic perspective of trace equivalence given by Hasuo in [18]. Therefore the same settings are assumed, *i.e.* we consider coalgebras for functors like TF where T and F are endofunctors over a category C with all finite limits and ω-colimits; (T, μ, η) is a monad; there exists a natural transformation λ distributing F over T; the Kleisli category $\mathrm{Kl}(T)$ is CPPO-enriched and has, for any $X \in$ C, a final $(\_ + X)$-coalgebra. Before describing the saturation construction let us state the main definition of this Section.

Definition 14. *Given two TF_τ-coalgebras (X, α) and (Y, β), a span of jointly monic arrows $X \xleftarrow{p} R \xrightarrow{q} Y$ describes a weak bisimulation between α and β if and only if there exists an epic cospan $X \xrightarrow{f} C \xleftarrow{g} Y$ such that (R, p, q) is the final span to make the following diagram commute:*

where α^w and β^w are the weak saturated TF_τ-coalgebras w.r.t. f and g.

Let us see how the weak saturation α^w is defined. In our setting, the traces of a TF-coalgebra α are described by the final map tr_α from the lifting of α in $\mathrm{Kl}(T)$ to the final $\overline{F}$-coalgebra where $\overline{F}$ is the lifting of F to $\mathrm{Kl}(T)$ induced by the distributive law $\lambda : FT \Longrightarrow TF$ (*cf.* [18]). Rawly speaking, the monad T can be thought as describing the branching of the system whereas the observables are characterized by F. Assuming this point of view, any F can be extended with silent actions τ as the free pointed functor

$$F_\tau \triangleq X + FX.$$

Now, a TF_τ-coalgebra α can be "determinized" by means of its *iterate* [36]:

$$\mathrm{itr}_\alpha \triangleq \nabla_{FX} \circ \mathrm{tr}_\alpha$$

where ∇ is the codiagonal; the traces refer to α seen as a $T(X + B)$ for $B = FX$ and (X, itr_α) is a TF-coalgebra. The iterate offers an elegant and general way to "compress" executions with leading silent transitions like $\tau^* a$ into single-step transitions with exactly one observable but retaining the effects of the entire execution within the monad T.

These results can be used to cover executions ending with an observable and hence do not directly lend themselves to equivalences based also on trailing silent actions like in the case $\tau^* a \tau^*$, as required by the weak bisimulation. However, let us suppose to have, for any given TF_τ-coalgebra (X, α), the T-coalgebra (X, α^τ) describing how each state reaches every class with τ-transition only; then, the coalgebra describing reachability by $\tau^* a \tau^*$ is exactly:

$$\alpha^\flat : X \xrightarrow{\text{itr}_\alpha} TFX \xrightarrow{TF\alpha^\tau} TFTX \xrightarrow{\lambda_X} TTFX \xrightarrow{\mu} TFX.$$

Then the saturated coalgebra α^w is defined by means of the 2-cell structure of $\text{KL}(T)$ as the join described by the diagram below.

$$
\begin{array}{c}
X \\
\alpha^\tau \nearrow \quad | \quad \nwarrow \alpha^\flat \\
X \xleftarrow{\quad} \sqsubseteq \quad \alpha^w \triangleq \sqcup \quad \sqsupseteq \xrightarrow{\quad} FX \\
\iota_1 \searrow \quad \downarrow \quad \swarrow \iota_2 \\
(X + FX)
\end{array}
\tag{10}
$$

This definition points out that τ^* and $\tau^* a \tau^*$ are two close but different cases.

In order to define the T-coalgebra α^τ, first we need to be able to consider only the silent action of the given α. This information can be isolated from α by means of the same structure used in (10). Therefore we define, for every TF_τ-coalgebra α, its silent and observable parts, namely α^s and α^o, as the (greatest) arrows to make the following diagram commute and have α as their join.

$$
\begin{array}{c}
X \\
\alpha^s \nearrow \quad | \quad \nwarrow \alpha^o \\
X \xleftarrow{\quad} \sqsubseteq \quad \alpha = \sqcup \quad \sqsupseteq \xrightarrow{\quad} FX \\
\iota_1 \searrow \quad \downarrow \quad \swarrow \iota_2 \\
(X + FX)
\end{array}
$$

Because α^τ has to describe how each class is reached, classes can be used as the observables needed to apply the iterate construction to α^τ. However, to be able to select the class to be reached and consider it as the only one observable by the iterate (likewise F_τ distinguish silent and observable actions by means of a coproduct) we need X and C to be represented as indexed coproducts of simpler canonical subobjects corresponding to the classes induced by $f : X \to C$. Henceforth, for simplicity we assume $X \cong X \cdot 1$ and $C \cong C \cdot 1$. For each class $c : 1 \to C$ let $X \cong \overline{X}_c + X_c$ be the split induced by c. This extends to the coalgebra α^s (by coproduct) determining the coalgebra: $\alpha^s_c : \overline{X}_c \to T(\overline{X}_c + X_c)$ whose iterate is the map $\alpha^+_c : \overline{X}_c \to T(X_c)$ describing executions of silent action only ending in c (but starting elsewhere). This yields a C-indexed family of morphisms which together describe τ^+ and the information is collected in one T-algebra as a join in the 2-cell like (10). For this join to be admissible we require T to not exceed the completeness of 2-cells, *i.e.* for any $x : 1 \to X \in \text{C}$ the supremum of the set $\text{KL}(T)(1, X)$ determined by $\text{KL}(T)(x, X)$ exists. Reworded if cells are κ-CPPOs, then T is κ-finitary; *e.g.* in SET T-coalgebras describe image κ-bounded T-branching systems. Thus, for every x the family of arrows $\{\alpha^+_c \circ x\}$ is limited by κ and can be joined. These are composed in α^+ as the universal arrow in the X-fold coproduct. The last step is provided by the monad unit which is a T-coalgebra describing how states reach their containing class and can be easily joined to the above obtaining, finally, α^τ. This completes the construction of α^w.

Weighted labelled transition systems Assuming at most countably many actions, image-countable $\mathfrak{W}$-WLTs are in 1-1 correspondence with $\mathcal{F}_\mathfrak{W}(A \times \_)$-coalgebras where $(\mathcal{F}_\mathfrak{W}, \mu, \eta)$ is the monad of $\mathfrak{W}$-valued functions with at most countable support. In particular, $\mathcal{F}_\mathfrak{W} X$ is the set morphisms from $\text{SET}(X, W)$ factoring through $\mathbb{N}$. On arrows $\mathcal{F}_\mathfrak{W}$ is defined as $\mathcal{F}_\mathfrak{W} f(\varphi)(y) \triangleq$

$\sum_{x \in f^{-1}(y)} \varphi(x)$. The unit η is defined as $\eta(x)(y) \triangleq 1$ for $x = y$ and 0 otherwise, and the multiplication μ is defined as $\mu(\psi)(x) \triangleq \sum_{\varphi} \psi(\varphi) \cdot \varphi(x)$.

If $\mathfrak{W}$ is the boolean semiring, $\mathcal{F}_{\mathfrak{W}}$ is precisely the countable powerset monad. Strength and double strength readily generalize to every $\mathcal{F}_{\mathfrak{W}}$ and by [19] there is a canonical law λ distributing $(A \times \_)$ over $\mathcal{F}_{\mathfrak{W}}$. The semiring $\mathfrak{W}$ can be easily endowed with an ordering which lifts point-wise to $\mathfrak{W}$-valued functions [17]. In particular, any ω-semiring with a natural order (*cf.* Section 3.2) yields a CPPO-enriched $\mathrm{KL}(\mathcal{F}_{\mathfrak{W}})$ with bottom the constantly 0 function.

Theorem 13. *Let T be $\mathcal{F}_{\mathfrak{W}}$ and F and $(A \times \_)$. For any given TF_τ-coalgebra α and its corresponding $\mathfrak{W}$-LTS, Definition 14 and Definition 11 coincide.*

Proof. By unfolding of Definition 14 and by minimality of executions considered by the construction of α^w. □

7 Conclusions and future work

In this paper we have introduced a general notion of *weak weighted bisimulation* which applies to any system that can be specified as a LTS weighted over a *semiring*. The semiring structure allows us to compositionally extend weights to multi-step transitions. We have shown that our notion of weak bisimulations naturally covers the cases of non-deterministic, fully-probabilistic, and stochastic systems, among others. We described a "universal" algorithm for computing weak bisimulations parametric in the underlying semiring structure and proved its decidability for every positively ordered ω-semiring. Finally, we gave a categorical account of the coalgebraic construction behind these results, providing the basis for extending the results presented here to other behavioural equivalences.

Our results came with a great flexibility offered, from one hand, by the possibility to instantiate WLTSs to several systems (by just providing opportune semirings) and, from the other, by the possibility to consider many other behavioural equivalences simply by changing the observation patterns used in Definition 11 and in the linear equations systems at the core of the proposed algorithm as described by Example 4.

A possible future work is to improve the efficiency of our algorithm, *e.g.* by extending Paige-Tarjan's algorithm for strong bisimulation instead of Kannellakis-Smolka's, or using more recent approaches based on symbolic bisimulations [43].

The algorithm presented is based on Kanellakis-Smolka's. A possible future work could be to improve the efficiency of this algorithm, *e.g.* by extending Paige-Tarjan's algorithm for strong bisimulation, or more recent approaches like symbolic bisimulations (*e.g.* [43, 13]).

Obviously, for specific systems and semirings there are solutions more efficient than our. For instance, in the case of systems over the semiring of non-negative real numbers (which captures *e.g.* and probabilistic systems) the asymptotic upper bound for time complexity of our algorithm is $\mathcal{O}(mn^{3.8})$ (since $\mathcal{L}_{\mathbb{R}_0^+}(n)$ is in $\mathcal{O}(n^{2.8})$ using [1]). However, deciding weak bisimulation for fully-probabilistic systems is in $\mathcal{O}(mn^3)$ on the worst case using the algorithm proposed by Baier and Hermanns in [3] (the original analysis assumed A to be fixed resulting in the worst case complexity $\mathcal{O}(n^3)$). Their algorithm capitalise on properties not available at the general level of WLTSs (even under the assumption of suitable orderings), such as: sums of outgoing transitions are bounded, there are complementary events, real numbers have more structure than a semiring, weak and delay bisimulations coincide for finite fully-probabilistic systems (*e.g.* this does not hold for non-deterministic LTSs). The aim of future work is to generalize the efficient results of [3] to a parametrized algorithm for constrained WLTSs, or at least for some classes of WLTSs subject to suitable families of constraints.

The construction presented in Section 6 introduces some techniques and tools that can be used to deal with other behavioural equivalences. In fact, we think that many behavioural equivalences can be obtained by "assembling" smaller components, by means of 2-splits, 2-merges and *iterate*, as we did for weak bisimulation. We plan to provide a formal, and easy to use, language for describing and combining these "building blocks" in a modular way.

An important direction for future work is to generalize our framework by weakening the assumptions on the underlying category (introduced in order to observe and manipulate equivalence classes) and by considering different behavioural functors. In particular, we intend to extend this framework to ULTRaSs, *i.e.* the generalization of WLTSs recently proposed by Bernardo et al. in [7]. These are an example of *staged transition systems*, where several behavioural functors (or stages) are "stacked" together.

References

[1] A. V. Aho, J. E. Hopcroft, and J. D. Ullman. *The Design and Analysis of Computer Algorithms*. Addison-Wesley Publishing Company, 1974.

[2] C. Baier. On algorithmic verification methods for probabilistic systems. Universität Mannheim, 1998.

[3] C. Baier, B. Engelen, and M. E. Majster-Cederbaum. Deciding bisimilarity and similarity for probabilistic processes. *J. Comput. Syst. Sci.*, 60(1):187–231, 2000.

[4] C. Baier and H. Hermanns. Weak bisimulation for fully probabilistic processes. In *CAV*, pages 119–130, 1997.

[5] M. Bernardo. Weak markovian bisimulation congruences and exact CTMC-level aggregations for concurrent processes. In *QAPL*, pages 122–136, 2012.

[6] M. Bernardo and A. Aldini. Weak markovian bisimilarity: Abstracting from prioritized/weighted internal immediate actions. In *ICTCS*, pages 39–56, 2007.

[7] 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.

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

[9] T. Brengos. Weak bisimulations for coalgebras over ordered functors. In J. C. M. Baeten, T. Ball, and F. S. de Boer, editors, *IFIP TCS*, volume 7604 of *Lecture Notes in Computer Science*, pages 87–103. Springer, 2012.

[10] T. Brengos. An algebraic approach to weak and delay bismulation in coalgebra. *CoRR*, abs/1302.4856, 2013.

[11] P. Buchholz and P. Kemper. Weak bisimulation for (max/+) automata and related models. *Journal of Automata, Languages and Combinatorics*, 8(2):187–218, 2003.

[12] P. R. D'Argenio and J.-P. Katoen. A theory of stochastic systems. part ii: Process algebra. *Information and Computation*, 203(1):39–74, 2005.

[13] A. Dovier, C. Piazza, and A. Policriti. An efficient algorithm for computing bisimulation equivalence. *Theor. Comput. Sci.*, 311(1-3):221–256, 2004.

[14] R. J. V. Glabbeek, S. A. Smolka, and B. Steffen. Reactive, generative and stratified models of probabilistic processes. *Information and Computation*, 121:130–141, 1990.

[15] N. Götz, U. Herzog, and M. Rettelbach. Multiprocessor and distributed system design: The integration of functional specification and performance analysis using stochastic process algebras. In *Performance Evaluation of Computer and Communication Systems, Joint Tutorial Papers of Performance '93 and Sigmetrics '93*, pages 121–146, London, UK, UK, 1993. Springer-Verlag.

[16] D. Grigoriev and V. V. Podolskii. Complexity of tropical and min-plus linear prevarieties. *CoRR*, abs/1204.4578, 2012.

[17] J. S. Han, H. S. Kim, and J. Neggers. Semiring orders in a semiring. *Appl. Math*, 6(1):99–102, 2012.

[18] I. Hasuo. Generic forward and backward simulations II: Probabilistic simulation. In P. Gastin and F. Laroussinie, editors, *CONCUR*, volume 6269 of *Lecture Notes in Computer Science*, pages 447–461. Springer, 2010.

[19] I. Hasuo, B. Jacobs, and A. Sokolova. Generic trace semantics via coinduction. *Logical Methods in Computer Science*, 3(4), 2007.

[20] H. Hermanns, U. Herzog, and J.-P. Katoen. Process algebra for performance evaluation. *Theoretical Computer Science*, 274(1-2):43–87, 2002.

[21] J. Hillston. *A compositional approach to performance modelling*. Cambridge University Press, New York, NY, USA, 1996.

[22] O. H. Jensen. *Mobile Processes in Bigraphs*. PhD thesis, University of Aalborg, 2008. To appear.

[23] P. C. Kanellakis and S. A. Smolka. Ccs expressions, finite state processes, and three problems of equivalence. *Inf. Comput.*, 86(1):43–68, 1990.

[24] J. G. Kemeny and J. L. Snell. *Finite Markov Chains*. Springer, 1976.

[25] B. Klin. Structural operational semantics for weighted transition systems. In J. Palsberg, editor, *Semantics and Algebraic Specification*, volume 5700 of *Lecture Notes in Computer Science*, pages 121–139. Springer, 2009.

[26] B. Klin and V. Sassone. Structural operational semantics for stochastic process calculi. In R. M. Amadio, editor, *FoSSaCS*, volume 4962 of *Lecture Notes in Computer Science*, pages 428–442. Springer, 2008.

[27] B. Klin and V. Sassone. Structural operational semantics for stochastic and weighted transition systems. *Information and Computation*, 2012.

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

[29] G. L. Litvinov, V. P. Maslov, A. Y. Rodionov, and A. N. Sobolevski. Universal algorithms, mathematics of semirings and parallel computations. In *Coping with Complexity: Model Reduction and Data Analysis*, pages 63–89. Springer, 2011.

[30] N. López and M. Núñez. Weak stochastic bisimulation for non-markovian processes. In *Proceedings of the Second international conference on Theoretical Aspects of Computing*, pages 454–468. Springer-Verlag, 2005.

[31] N. López, M. Núñez, and F. Rubio. An integrated framework for the performance analysis of asynchronous communicating stochastic processes. *Formal Aspects of Computing*, 16(3):238–262, 2004.

[32] R. Milner. *Communication and Concurrency*. Prentice-Hall, 1989.

[33] R. Nicola, D. Latella, M. Loreti, and M. Massink. Rate-based transition systems for stochastic process calculi. In *Proceedings of the 36th Internatilonal Collogquium on Automata, Languages and Programming: Part II*, ICALP '09, pages 435–446. Springer-Verlag, 2009.

[34] P. Panangaden. *Labelled Markov Processes*. Imperial College Press, London, U.K., 2009.

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

[36] A. Silva and B. Westerbaan. A coalgebraic view of ε-transitions. In *CALCO*, 2013.

[37] P. Sobociński. Relational presheaves as labelled transition systems. In *Coalgebraic Methods in Computer Science (CMCS '12)*, volume 7399 of *LNCS*, pages 40–50. Springer, 2012.

[38] A. Sokolova, E. P. de Vink, and H. Woracek. Weak bisimulation for action-type systems. In *Proc. CTCS'04*, pages 211–228. ENTCS 122, 2005.

[39] A. Sokolova, E. P. de Vink, and H. Woracek. Coalgebraic weak bisimulation for action-type systems. *Scientific Annals of Computer Science*, 19:93–144, 2009.

[40] C. Tofts. A synchronous calculus of relative frequency. In *CONCUR'90 Theories of Concurrency: Unification and Extension*, pages 467–480. Springer, 1990.

[41] R. J. van Glabbeek. The linear time—branching time spectrum. In *CONCUR*, pages 278–297, 1990.

[42] R. J. van Glabbeek. The linear time—branching time spectrum II. In *CONCUR*, pages 66–81, 1993.

[43] R. Wimmer, M. Herbstritt, H. Hermanns, K. Strampp, and B. Becker. Sigref–a symbolic bisimulation tool box. In *Automated Technology for Verification and Analysis*, pages 477–492. Springer, 2006.

[44] A. C. Zaanen. *An introduction to the theory of integration*. Amsterdam, 1958.