Available online at www.sciencedirect.com

SCIENCE DIRECT

Electronic Notes in Theoretical Computer Science

ELSEVIER Electronic Notes in Theoretical Computer Science 106 (2004) 63–90

www.elsevier.com/locate/entcs

On Logics for Coalgebraic Simulation

Corina Cîrstea[1]

School of Electronics and Computer Science
University of Southampton

Abstract

We investigate logics for coalgebraic simulation from a compositional perspective. Specifically, we show that the expressiveness of an inductively-defined language for coalgebras w.r.t. a given notion of simulation comes as a consequence of an expressivity condition between the language constructor used to define the language for coalgebras, and the relator used to define the notion of simulation. This result can be instantiated to obtain Baltag's logics for coalgebraic simulation, as well as a logic which captures simulation on unlabelled probabilistic transition systems. Moreover, our approach is compositional w.r.t. coalgebraic types. This allows us to derive logics which capture other notions of simulation, including trace inclusion on labelled transition systems, and simulation on discrete Markov processes.

Keywords: coalgebra, simulation, modal logic

1 Introduction

Simulations are widely used in computer science, typically to prove refinement relations between dynamical systems. The connection between simulations and coalgebra was probably first noted in [13] (see also [8]), where the objective was to prove refinement relations between recursively-defined programs. This connection was further investigated in [1], where logics capturing simulation were also studied. Additional properties of coalgebraic simulations, including a characterization of the similarity relation on the final coalgebra, were subsequently proved in [10].

The method used in [1] to define logics for simulation builds on, and at the same time generalizes the approach described in [12] for defining expressive

[1] Email: cc2@ecs.soton.ac.uk

doi:10.1016/j.entcs.2004.02.026

logics for bisimulation. The resulting logics are generic in coalgebraic types, and employ a single modal operator derived directly from the coalgebraic signature. These logics are, however, difficult to use in actual specification, as their syntax does not reflect the structure of the underlying types.

The present paper describes a compositional method for defining logics which capture simulation. This method allows logics for combinations of coalgebraic types to be derived from logics for the types being combined. Thus, the structure of the underlying types is reflected in the modal operators employed by the resulting logics.

A similar approach to defining logics was taken in [5], where logics capturing bisimulation were investigated from a compositional perspective. Specifically, it was shown in [5] that the expressiveness w.r.t. bisimulation of an inductively-defined language for coalgebras follows from an expressivity condition referring to one step in the definition of the language. In the case of logics for simulation, the situation is more complex. On the one hand, ways to combine notions of simulation for different coalgebraic types are needed. On the other hand, the sought logics must be tailored to particular notions of simulation, and therefore the expressivity condition used in [5] must be adapted accordingly.

The paper is structured as follows. In Section 2, we recall the coalgebraic approach to defining simulation. In Section 3, we provide an alternative characterization of monotonic relators, the concept underlying the definition of coalgebraic simulation [13,8], and use this characterization to define a notion of simulation for unlabelled probabilistic transition systems. Next, in Section 4, we propose an inductive method for defining logics which capture simulation, much in the spirit of [5]. Using this method, the expressiveness of a logic for simulation comes as a consequence of an expressivity condition between a language constructor and a monotonic relator. This method can be applied to obtain the logics defined in [1], as well as a logic capturing simulation on unlabelled probabilistic transition systems. Finally, in Section 5, we show that our method for defining logics for simulation is compositional w.r.t. coalgebraic types. Operations on coalgebraic types, including functor composition, product, coproduct and exponentiation are shown to induce corresponding operations on monotonic relators on the one hand, and on language constructors on the other. Moreover, the resulting operations are shown to preserve the previously-mentioned expressivity condition. This allows us to derive logics which capture trace inclusion on labelled transition systems and simulation on discrete Markov processes, respectively. In the latter case, the logic obtained is essentially the logic considered in [7]. Thus, in this case, we obtain both a coalgebraic characterization of simulation on discrete Markov processes, and an alternative proof of expressiveness of the logic in [7] w.r.t. simulation.

2 Preliminaries

Here we fix the notation for subsequent sections, recall some basic definitions and results concerning relations and respectively coalgebras, and summarize the coalgebraic approach to defining simulation.

2.1 Relations

We write Rel for the category having objects given by tuples $\langle A, B, R\rangle$ with $R \subseteq A \times B$, and arrows from $\langle A, B, R\rangle$ to $\langle C, D, S\rangle$ given by pairs $\langle f, g\rangle$ with $f : A \to C$ and $g : B \to D$ such that $(f \times g)(R) \subseteq S$.

Remark 2.1 This is not the only way of defining a category of relations. One can also consider the category having, as objects, pairs consisting of a set and a binary relation on it, and as arrows, functions between sets which preserve the relations. Yet another possibility is to consider the category having sets as objects and relations as arrows. All these categories are usually denoted Rel. Our own definition of Rel follows [10].

Given a relation $R \subseteq A \times B$, we write π_1^R and π_2^R for $\pi_1 \circ \iota : R \to A$ and $\pi_2 \circ \iota : R \to B$, respectively, where $\pi_1 : A \times B \to A$ and $\pi_2 : A \times B \to B$ are the product projections, and where $\iota : R \to A \times B$ is the inclusion map. Also, we write R^{op} for the converse of a relation R, and $\mathsf{Gr} f \subseteq A \times B$ for the relation defining the graph of a function $f : A \to B$. The composition of relations $R \subseteq A \times B$ and $S \subseteq B \times C$ is denoted $S \circ R \subseteq A \times C$.

We let $\mathsf{U} : \mathsf{Rel} \to \mathsf{Set} \times \mathsf{Set}$ denote the functor taking relations to the underlying sets. Then, U is a *fibration* [2] . For, given $f : A \to C$, $g : B \to D$ and $S \subseteq C \times D$, letting $a\, R\, b$ if and only if $f(a)\, S\, g(b)$ makes $\langle f, g\rangle : \langle A, B, R\rangle \to \langle C, D, S\rangle$ a cartesian map. The cartesian maps of U are thus the relation-reflecting maps in Rel.

We also let Preord denote the category of preorders and monotonic maps. Then, Preord is (isomorphic to) a sub-category of Rel. Moreover, if $\mathsf{V} : \mathsf{Preord} \to \mathsf{Set}$ takes preorders to the underlying sets, then V is a fibration. The cartesian maps of V are the order-reflecting maps in Preord.

The following also holds:

Proposition 2.2 Rel *and* Preord *are complete categories.*

[2] See [3] for a definition of this notion.

Limits in Rel and Preord are constructed from limits in Set and limits in certain fibres of U and V, respectively.

2.2 Coalgebras

For an endofunctor $\mathsf{T} : \mathsf{C} \to \mathsf{C}$, a T-*coalgebra* is a pair $\langle C, \gamma \rangle$ with $\gamma : C \to \mathsf{T}C$ a C-arrow. Also, a T-*coalgebra homomorphism* from $\langle C, \gamma \rangle$ to $\langle D, \delta \rangle$ is a C-arrow $f : C \to D$ such that $\mathsf{T}f \circ \gamma = \delta \circ f$. In what follows, we will consider coalgebras over the categories Set, Rel and Preord.

Example 2.3 A-labelled, image-finite transition systems can be modelled as coalgebras of the functor $(\mathcal{P}_\omega)^A$, where $\mathcal{P}_\omega : \mathsf{Set} \to \mathsf{Set}$ takes a set to the set of its finite subsets and a function to its direct image, and X^A denotes the set of functions $A \to X$. The functors $\mathcal{P}_\omega$ and $(\mathcal{P}_\omega)^A$ preserve weak pullbacks and are ω-*accessible* [3].

Example 2.4 A-labelled probabilistic transition systems can be modelled as coalgebras of the functor $(1 + \mathcal{D}_\omega)^A : \mathsf{Set} \to \mathsf{Set}$, where $\mathcal{D}_\omega : \mathsf{Set} \to \mathsf{Set}$ is the *finite probability distribution functor*, defined by:

$$\mathcal{D}_\omega X = \{\, \mu : X \to [0,1] \mid \mathsf{supp}(\mu) \text{ finite} \,,\ \sum_{x \in X} \mu(x) = 1 \,\} \quad \text{for } X \in |\mathsf{Set}|$$

with $\mathsf{supp}(\mu) = \{\, x \in X \mid \mu(x) \neq 0 \,\}$ for $\mu : X \to [0,1]$, and:

$$(\mathcal{D}_\omega f)(\mu)(y) = \mu[f^{-1}(\{y\})] \quad \text{for } f : X \to Y \,,\ \mu \in \mathcal{D}_\omega X \,,\ \text{and } y \in Y$$

with $\mu[Z] = \sum_{x \in Z} \mu(x)$ for $\mu : X \to [0,1]$ and $Z \subseteq X$. The functor $\mathcal{D}_\omega$ preserves weak pullbacks (see e.g. [12]), and so do the functors $1 + \mathcal{D}_\omega$ and $(1 + \mathcal{D}_\omega)^A$. Also, all these functors are ω-accessible.

Given $\mathsf{T} : \mathsf{Set} \to \mathsf{Set}$, a T-*bisimulation* between T-coalgebras $\langle C, \gamma \rangle$ and $\langle D, \delta \rangle$ is a relation $R \subseteq C \times D$ carrying a T-coalgebra structure $\rho : R \to \mathsf{T}R$ which makes $\pi_1^R : R \to C$ and $\pi_2^R : R \to D$ T-coalgebra homomorphisms. The largest T-bisimulation between $\langle C, \gamma \rangle$ and $\langle D, \delta \rangle$ is called T-*bisimilarity* and is denoted $\simeq$.

Example 2.5 A notion of bisimulation equivalence for probabilistic transition systems was defined in [11]. Moreover, it was shown in [6] that this notion is essentially the same as $(1 + \mathcal{D}_\omega)^A$-bisimulation. The following characterization of $1 + \mathcal{D}_\omega$-bisimulation was also given in [6]: a relation $R \subseteq C \times D$

[3] For a regular cardinal κ, an endofunctor is κ-*accessible* if it preserves κ-filtered colimits.

is a $1 + \mathcal{D}_\omega$-bisimulation between $\langle C, \gamma\rangle$ and $\langle D, \delta\rangle$ if and only if $c\,R\,d$ implies $\gamma(c)[X] = \delta(d)[Y]$ [4] for any $X \subseteq C$ and $Y \subseteq D$ such that $(\pi_1^R)^{-1}(X) = (\pi_2^R)^{-1}(Y)$.

For an endofunctor $\mathsf{T} : \mathsf{C} \to \mathsf{C}$ on a complete category, the *final sequence of* T is an ordinal-indexed sequence (Z_α) of C-objects, together with a family $(p^\alpha_\beta : Z_\alpha \to Z_\beta)_{\beta \leq \alpha}$ of C-arrows, subject to the following conditions:

(i) $Z_{\alpha+1} = \mathsf{T}Z_\alpha$

(ii) $p^{\alpha+1}_{\beta+1} = \mathsf{T}p^\alpha_\beta$ for $\beta \leq \alpha$

(iii) $p^\alpha_\alpha = 1_{Z_\alpha}$

(iv) $p^\alpha_\gamma = p^\beta_\gamma \circ p^\alpha_\beta$ for $\gamma \leq \beta \leq \alpha$

(v) if α is a limit ordinal, the cone $Z_\alpha, (p^\alpha_\beta)_{\beta<\alpha}$ for $(p^\beta_\gamma)_{\gamma\leq\beta<\alpha}$ is limiting.

The final sequence of T is uniquely defined by these conditions.

Remark 2.6 Given $\mathsf{T} : \mathsf{C} \to \mathsf{C}$ as above, one can define, for each T-coalgebra $\langle C, \gamma\rangle$, a cone $(\gamma_\alpha : C \to Z_\alpha)$ over the final sequence of T:

- $\gamma_\alpha = \mathsf{T}\gamma_\beta \circ \gamma$, if $\alpha = \beta + 1$;
- γ_α is the unique C-arrow satisfying $p^\alpha_\beta \circ \gamma_\alpha = \gamma_\beta$ for each $\beta < \alpha$, if α is a limit ordinal.

Then, T-coalgebra homomorphisms $f : \langle C, \gamma\rangle \to \langle D, \delta\rangle$ define morphisms of cones $f : (\gamma_\alpha : C \to Z_\alpha) \to (\delta_\alpha : D \to Z_\alpha)$. That is, $\delta_\alpha \circ f = \gamma_\alpha$ for any α.

Under some mild constraints on C and T, the final sequence of T can be used to construct a final T-coalgebra.

Proposition 2.7 ([14]) *If* $\mathsf{T} : \mathsf{C} \to \mathsf{C}$ *is an accessible endofunctor on a locally presentable category* [5]*, and if* T *preserves monics, then the final sequence of* T *stabilizes at some* α [6]*, and moreover,* Z_α *is the carrier of a final* T*-coalgebra.*

Moreover, in the case of ω-accessible endofunctors on Set, the cardinal α of Proposition 2.7 is at most $\omega + \omega$.

Proposition 2.8 ([14]) *If* $\mathsf{T} : \mathsf{Set} \to \mathsf{Set}$ *is* ω*-accessible, then the map* $p^{\omega+\omega+1}_{\omega+\omega} : Z_{\omega+\omega+1} \to Z_{\omega+\omega}$ *is an isomorphism, whereas the maps* $p^{\omega+n+1}_{\omega+n} : Z_{\omega+n+1} \to Z_{\omega+n}$ *with* $n = 0, 1, \ldots$ *are all injective.*

---

[4] By convention, $\gamma(c)[X] = 0$ if $\gamma(c) \in \iota_1(1)$.

[5] Each of the categories Set, Rel and Preord are locally ω-presentable.

[6] That is, $p^{\alpha+1}_\alpha : Z_{\alpha+1} \to Z_\alpha$ is an isomorphism.

2.3 Simulations

Notions of simulation between coalgebras have been studied in [13,8,1,10]. A summary of these approaches is given in the following. For this, we fix an endofunctor $\mathsf{T} : \mathsf{Set} \to \mathsf{Set}$.

The concept which lies at the heart of defining simulations is that of a relator. A (T-)*relator* [13] is a mapping from relations to relations, taking relations on $A \times B$ to relations on $\mathsf{T}A \times \mathsf{T}B$. A *monotonic* ($\mathsf{T}$-)*relator* [13] is required to satisfy some additional constraints, including preservation of inclusions between relations and preservation of relational composition. These constraints result in monotonic T-relators being essentially the same as endofunctors $\Gamma : \mathsf{Rel} \to \mathsf{Rel}$ satisfying:

(i) $\mathsf{U} \circ \Gamma = (\mathsf{T} \times \mathsf{T}) \circ \mathsf{U}$;

(ii) $=_{\mathsf{T}A} \subseteq \Gamma(=_A)$;

(iii) $\Gamma(S \circ R) = \Gamma(S) \circ \Gamma(R)$ for any $R \subseteq A \times B$ and $S \subseteq B \times C$.

In the sequel, we will identify monotonic relators with such endofunctors.

A generic example of a relator is the *minimal relator induced by* T [13], denoted $\Gamma_m : \mathsf{Rel} \to \mathsf{Rel}$, and defined by:

$$\Gamma_m(R) = \langle \mathsf{T}\pi_1^R, \mathsf{T}\pi_2^R \rangle (\mathsf{T}R) \subseteq \mathsf{T}A \times \mathsf{T}B \text{ for } R \subseteq A \times B$$

The minimal relator induced by T is monotonic if and only if T preserves weak pullbacks. Although not explicitly stated in [13], this observation is an immediate consequence of the results in [13, Section 2.2]. Irrespective of the preservation of weak pullbacks by T, the minimal relator is contained in any monotonic relator Γ, that is, $\Gamma_m(R) \subseteq \Gamma(R)$ for any relation R. Moreover, any monotonic relator Γ can be defined in terms of its action on equality relations and of Γ_m:

$$\Gamma(R) = \Gamma(=_B) \circ \Gamma_m(R) \circ \Gamma(=_A) \text{ for any } R \subseteq A \times B \tag{1}$$

Given a T-relator $\Gamma : \mathsf{Rel} \to \mathsf{Rel}$, the *transposed relator* $\Gamma^{\sim}$ takes a relation $R \subseteq A \times B$ to the relation $(\,\Gamma(R^{\mathsf{op}})\,)^{\mathsf{op}} \subseteq \mathsf{T}A \times \mathsf{T}B$.

Example 2.9 The minimal $\mathcal{P}_\omega$-relator $\Gamma_m : \mathsf{Rel} \to \mathsf{Rel}$ takes a relation $R \subseteq A \times B$ to the relation $\Gamma_m(R) \subseteq \mathcal{P}_\omega A \times \mathcal{P}_\omega B$ defined by:

$$X\, \Gamma_m(R)\, Y \quad \text{iff} \quad (\forall\, x \in X. \exists\, y \in Y.\, x\, R\, y \text{ and } \forall\, y \in Y. \exists\, x \in X.\, x\, R\, y)$$

for $X \in \mathcal{P}_\omega A$, $Y \in \mathcal{P}_\omega B$. Another $\mathcal{P}_\omega$-relator $\Gamma_\supseteq : \mathsf{Rel} \to \mathsf{Rel}$ can be defined by:

$$X\, \Gamma_\supseteq(R)\, Y \quad \text{iff} \quad \forall\, y \in Y. \exists\, x \in X.\, x\, R\, y$$

Both Γ_m and $\Gamma_\supseteq$ are monotonic relators. Moreover, $\Gamma_\supseteq(R) = \supseteq_B \circ \Gamma_m(R) \circ \supseteq_A$, where $\supseteq_A$ and $\supseteq_B$ are the containment relations on $\mathcal{P}_\omega A$ and $\mathcal{P}_\omega B$, respectively. We also note that $\Gamma_\supseteq$ preserves monics and is ω-accessible. (This observation will be used later in the paper.) Finally, the transposed relator $\Gamma_\subseteq = (\Gamma_\supseteq)^\sim$ is given by:

$$X\,\Gamma_\subseteq(R)\,Y \quad \text{iff} \quad \forall\, x \in X. \exists\, y \in Y.\, x\,R\,y$$

[13] also shows the existence of a one-to-one correspondence between monotonic relators and so-called *monotonic extensions of* T. These are functors $\sqsupseteq : \mathsf{Set} \to \mathsf{Preord}$ such that:

(i) $\mathsf{V} \circ \sqsupseteq = \mathsf{T}$;

(ii) if $A \subseteq B$ then $u \sqsupseteq_A v$ iff $u \sqsupseteq_B v$ for any $u, v \in \mathsf{T}A$;

(iii) (**monotonicity**) the following holds for $f : A \to C$, $g : B \to C$, $u \in \mathsf{T}A$ and $v \in \mathsf{T}B$:

$$(\mathsf{T}f)(u) \sqsupseteq (\mathsf{T}g)(v) \;\Rightarrow\; u\,(\Gamma_\sqsupseteq\{\,(a,b) \in A \times B \mid f(a) = g(b)\,\})\,v \qquad (2)$$

where $\Gamma_\sqsupseteq : \mathsf{Rel} \to \mathsf{Rel}$ denotes the *relator induced by* $\sqsupseteq$, defined by:

$$\Gamma_\sqsupseteq(R) = \sqsupseteq_B \circ \Gamma_m(R) \circ \sqsupseteq_A \;\text{ for } R \subseteq A \times B$$

Monotonic extensions induce monotonic relators, and moreover, any monotonic relator Γ arises from a unique monotonic extension $\sqsupseteq_\Gamma$, given by:

$$\sqsupseteq_{\Gamma,A} = \Gamma(=_A) \text{ for } A \in |\mathsf{Set}| \qquad (3)$$

Finally, any monotonic relator Γ restricts to an endofunctor on Preord, itself denoted Γ.

Example 2.10 The functor $\supseteq : \mathsf{Set} \to \mathsf{Preord}$ taking a set A to the containment relation $\supseteq_A$ on $\mathcal{P}_\omega A$ defines a monotonic extension of $\mathcal{P}_\omega$. The corresponding monotonic relator is $\Gamma_\supseteq$, as defined in Example 2.9.

The following is a reformulation of the definition of simulation given in [13] (see also [10]).

Definition 2.11 Let $\Gamma : \mathsf{Rel} \to \mathsf{Rel}$ be a monotonic relator. A **Γ-simulation** between T-coalgebras $\langle C, \gamma \rangle$ and $\langle D, \delta \rangle$ is a Γ-coalgebra $\langle \langle C, D, R \rangle, \langle \gamma, \delta \rangle \rangle$. The largest Γ-simulation between $\langle C, \gamma \rangle$ and $\langle D, \delta \rangle$ is called **Γ-similarity** and is denoted $\gtrsim$. If $c \in C$, $d \in D$ are such that $c \gtrsim d$, we say that *c simulates d*.

A Γ-simulation between $\langle C, \gamma \rangle$ and $\langle D, \delta \rangle$ is thus given by a relation $R \subseteq C \times D$ such that $c\,R\,d$ implies $\gamma(c)\;\Gamma(R)\;\delta(d)$ for any $c \in C$ and $d \in D$.

By taking the relator Γ of Definition 2.11 to be the minimal relator induced by T, we recover the definition of a T-bisimulation: a relation $R \subseteq C \times D$ is a T-bisimulation between $\langle C, \gamma \rangle$ and $\langle D, \delta \rangle$ if $\gamma(c)\ \langle \mathsf{T}\pi_1^R, \mathsf{T}\pi_2^R \rangle (\mathsf{T}R)\ \delta(d)$ holds whenever $c\,R\,d$.

Example 2.12 Let Γ_m and $\Gamma_\supseteq$ be as in Example 2.9. Then, Γ_m-simulations are the same as $\mathcal{P}_\omega$-bisimulations. Also, a relation $R \subseteq C \times D$ is a $\Gamma_\supseteq$-simulation between $\mathcal{P}_\omega$-coalgebras $\langle C, \gamma \rangle$ and $\langle D, \delta \rangle$ if, whenever $c\,R\,d$ and $d' \in \delta(d)$, there exists $c' \in \gamma(c)$ such that $c'\,R\,d'$.

Remark 2.13 A notion of *weak monotonic relator* was also defined in [1], based on ideas from [13]. This notion is similar to that of a monotonic relator, only in [1] a different category of relations, having sets as objects and relations as arrows, was considered. In this setting, the notion of relator does not depend on an endofunctor $\mathsf{T} : \mathsf{Set} \to \mathsf{Set}$. Instead, the fact that Set is a subcategory of the above-mentioned category of relations can be used to define what it means for a weak monotonic relator to *extend* an endofunctor T. A result in [4] then shows that a minimal relator extending T exists precisely when T preserves weak pullbacks. A notion of simulation induced by a weak monotonic relator was also defined in [1]. This notion is essentially the same as that of Definition 2.11. However, since the two definitions involve different categories of relations, it is not possible to directly transfer results between the two approaches.

In [10], functors $\sqsupseteq : \mathsf{Set} \to \mathsf{Preord}$ satisfying $\mathsf{V} \circ \sqsupseteq = \mathsf{T}$ were taken as primitive, and *lax relation lifting functors* $\mathsf{Rel}_\sqsupseteq(\mathsf{T}) : \mathsf{Rel} \to \mathsf{Rel}$, defined similarly to the relators $\Gamma_\sqsupseteq$, were considered. The difference w.r.t. [13] is that only the first condition in the definition of monotonic extensions was required of the functors $\sqsupseteq : \mathsf{Set} \to \mathsf{Preord}$. As a result, the induced lax relation lifting functors are not necessarily monotonic relators. However, once monotonicity is assumed, the setting of [10] coincides with that of [13].

It is shown in [10] that monotonicity of a relator Γ results in Γ-similarity enjoying some nice properties.

Proposition 2.14 ([10]) *The following hold for a monotonic relator* $\Gamma : \mathsf{Rel} \to \mathsf{Rel}$*:*

(i) Γ*-similarity on a* T*-coalgebra* $\langle C, \gamma \rangle$ *is a preorder on* C*;*

(ii) *given* T*-coalgebra homomorphisms* $f : \langle A, \alpha \rangle \to \langle B, \beta \rangle$ *and* $g : \langle C, \gamma \rangle \to \langle D, \delta \rangle$*,* $a \gtrsim c$ *if and only if* $f(a) \gtrsim g(c)$*, for* $a \in A$ *and* $c \in C$*;*

(iii) *similarity on the final* T*-coalgebra is the final* Γ*-coalgebra.*

Remark 2.15 By taking f and g in (ii) of Proposition 2.14 to be the unique homomorphisms $!_\alpha : \langle A, \alpha\rangle \to \langle Z, \zeta\rangle$ and $!_\gamma : \langle C, \gamma\rangle \to \langle Z, \zeta\rangle$ into the final T-coalgebra, we obtain that Γ-similarity between $\langle A, \alpha\rangle$ and $\langle C, \gamma\rangle$ is the domain of the cartesian map $\langle !_\alpha, !_\gamma\rangle$ induced by the Γ-similarity relation on the final T-coalgebra. This observation, together with (iii) of Proposition 2.14, will later allow us to define logics which capture Γ-similarity.

3 Monotonic Relators Revisited

Here we give an alternative characterization of monotonic relators. This characterization will prove more convenient for our purposes; in particular, it will allow us to define a notion of simulation for probabilistic transition systems. The alternative characterization has a more categorical flavour than the original definition, as it replaces the preservation of relational composition by a monotonic relator by preservation of a property of arrows in Rel.

Proposition 3.1 *Let* $\mathsf{T} : \mathsf{Set} \to \mathsf{Set}$*, and let* $\Gamma : \mathsf{Rel} \to \mathsf{Rel}$ *be such that:*

(i) $\mathsf{U} \circ \Gamma = (\mathsf{T} \times \mathsf{T}) \circ \mathsf{U}$*;*

(ii) $=_{\mathsf{T}A} \subseteq \Gamma(=_A)$*.*

Then, Γ *is a monotonic relator if and only if* Γ *preserves cartesian maps.*

Proof. Any monotonic relator Γ is uniquely determined by its induced monotonic extension $\sqsupseteq_\Gamma$, defined by (3). It therefore suffices to prove that, in the presence of (i) and (ii) above, condition (2) of Section 2.3 is equivalent to the preservation by Γ of cartesian maps.

We begin by noting that (2) is equivalent to Γ preserving cartesian maps of form $\langle f, g\rangle : \langle A, B, R\rangle \to \langle C, C, =_C\rangle$. Thus, one half of the previously-mentioned equivalence follows immediately. To prove the other half, assume that Γ is a monotonic relator. Then, observe that by taking $g = 1_C$ and respectively $f = 1_C$ in (2), we obtain:

$$\Gamma\mathsf{Gr}(f) = \sqsupseteq_{\Gamma,C} \circ \mathsf{Gr}(\mathsf{T}f) \qquad\qquad \Gamma(\mathsf{Gr}(g)^{\mathsf{op}}) = \mathsf{Gr}(\mathsf{T}g)^{\mathsf{op}} \circ \sqsupseteq_{\Gamma,C} \tag{4}$$

Now let $\langle f, g\rangle : \langle A, B, R\rangle \to \langle C, D, S\rangle$ be a cartesian map. Thus, $R = \mathsf{Gr}(g)^{\mathsf{op}} \circ S \circ \mathsf{Gr}(f)$. The fact that $\langle \mathsf{T}f, \mathsf{T}g\rangle : \langle \mathsf{T}A, \mathsf{T}B, \Gamma R\rangle \to \langle \mathsf{T}C, \mathsf{T}D, \Gamma S\rangle$ is itself a cartesian map, i.e. $\Gamma R = \mathsf{Gr}(\mathsf{T}g)^{\mathsf{op}} \circ \Gamma S \circ \mathsf{Gr}(\mathsf{T}f)$, follows from:

$$
\begin{array}{rcl}
\Gamma R & = & \\
\Gamma(\mathsf{Gr}(g)^{\mathsf{op}}) \circ \Gamma S \circ \Gamma(\mathsf{Gr}(f)) & = & (4) \\
\mathsf{Gr}(\mathsf{T}g)^{\mathsf{op}} \circ \sqsupseteq_{\Gamma,D} \circ \Gamma S \circ \sqsupseteq_{\Gamma,C} \circ \mathsf{Gr}(\mathsf{T}f) & = & (1) \\
\mathsf{Gr}(\mathsf{T}g)^{\mathsf{op}} \circ \sqsupseteq_{\Gamma,D} \circ \sqsupseteq_{\Gamma,D} \circ \Gamma_m S \circ \sqsupseteq_{\Gamma,C} \circ \sqsupseteq_{\Gamma,C} \circ \mathsf{Gr}(\mathsf{T}f) & = & \\
\mathsf{Gr}(\mathsf{T}g)^{\mathsf{op}} \circ \sqsupseteq_{\Gamma,D} \circ \Gamma_m S \circ \sqsupseteq_{\Gamma,C} \circ \mathsf{Gr}(\mathsf{T}f) & = & (1) \\
\mathsf{Gr}(\mathsf{T}g)^{\mathsf{op}} \circ \Gamma S \circ \mathsf{Gr}(\mathsf{T}f) & &
\end{array}
$$

The first of the above equalities uses the preservation of relational composition by Γ, whereas the fourth equality exploits the fact that $\sqsupseteq_{\Gamma,C}$ and $\sqsupseteq_{\Gamma,D}$ are preorders. Hence, Γ preserves cartesian maps. This concludes the proof. □

Thus, monotonic relators can alternatively be defined as functors satisfying (i) and (ii) of Proposition 3.1 and preserving cartesian maps. This observation will be used extensively in what follows.

Remark 3.2 It is also possible to give a fully categorical characterization of monotonic relators, namely by replacing condition (ii) of Proposition 3.1 by the requirement that Γ restricts to an endofunctor on Preord. However, for the purposes of this paper, the characterization provided by Proposition 3.1 is the most useful one.

Remark 3.3 The proof of Proposition 3.1 also gives:

$$\Gamma(\mathsf{Gr}(g)^{\mathsf{op}}) \circ \Gamma S = \mathsf{Gr}(\mathsf{T}g)^{\mathsf{op}} \circ \Gamma S \qquad \Gamma S \circ \Gamma(\mathsf{Gr}(f)) = \Gamma S \circ \mathsf{Gr}(\mathsf{T}f)$$

for any $f : A \to C$, $g : B \to D$ and $S \subseteq C \times D$.

Since all the relators considered in the following are monotonic, from now on we will simply use the term (T-)*relator* to refer to a monotonic (T-)relator.

We now define a relator for probabilistic transition systems, and investigate the notion of simulation induced by this relator.

3.1 Probabilistic Simulation

In defining a notion of simulation for unlabelled probabilistic transition systems (modelled as $1+\mathcal{D}_\omega$-coalgebras), it will prove convenient to work with an endofunctor slightly more general than $1 + \mathcal{D}_\omega$. Specifically, we will consider the *finite sub-probability distribution functor* $\mathcal{S}_\omega : \mathsf{Set} \to \mathsf{Set}$, defined by:

$$\mathcal{S}_\omega X = \{ \mu : X \to [0,1] \mid \mathsf{supp}(\mu) \text{ finite },\ \sum_{x \in X} \mu(x) \leq 1 \} \quad \text{for } X \in |\mathsf{Set}|$$

$$(\mathcal{S}_\omega f)(\mu)(y) = \mu[f^{-1}(\{y\})] \quad \text{for } f : X \to Y\ ,\ \mu \in \mathcal{S}_\omega X\ ,\ \text{and } y \in Y.$$

The coalgebraic type $\mathcal{S}_\omega$ is a generalization of the coalgebraic type $1 + \mathcal{D}_\omega$, in a sense made precise in the following.

Remark 3.4 Any $1 + \mathcal{D}_\omega$-coalgebra can be regarded as an $\mathcal{S}_\omega$-coalgebra. To see this, let $\eta : 1 + \mathcal{D}_\omega \Rightarrow \mathcal{S}_\omega$ be the natural transformation given by:

$$\eta_X(\iota_1(*))(x) = 0 \text{ for } x \in X$$
$$\eta_X(\iota_2(\mu)) = \mu$$

with $X \in |\mathsf{Set}|$. Then, η induces a functor $\mathsf{U}_\eta : \mathsf{Coalg}(1 + \mathcal{D}_\omega) \to \mathsf{Coalg}(\mathcal{S}_\omega)$, which takes a $1 + \mathcal{D}_\omega$-coalgebra $\langle C, \gamma \rangle$ to the $\mathcal{S}_\omega$-coalgebra $\langle C, \eta_C \circ \gamma \rangle$.

By using $\mathcal{S}_\omega$ to model unlabelled probabilistic transition systems, we provide a unified treatment of terminal states (i.e. states for which no transition is possible) and non-terminal ones.

Proposition 3.5 *$\mathcal{S}_\omega$ preserves weak pullbacks and is ω-accessible.*

An $\mathcal{S}_\omega$-relator can now be defined by relaxing the conditions in the characterization of $1 + \mathcal{D}_\omega$-bisimulation (see Example 2.5).

Definition 3.6 Let $\Gamma_\omega : \mathsf{Rel} \to \mathsf{Rel}$ be such that:

- Γ_ω takes $R \subseteq A \times B$ to $\Gamma_\omega R \subseteq \mathcal{S}_\omega A \times \mathcal{S}_\omega B$, where $\mu\,(\Gamma_\omega R)\,\nu$ if and only if $\mu[X] \geq \nu[Y]$ for any $X \subseteq A$ and $Y \subseteq B$ such that $(\pi_1^R)^{-1}(X) \supseteq (\pi_2^R)^{-1}(Y)$;
- Γ_ω takes $\langle f, g \rangle : \langle A, B, R \rangle \to \langle C, D, S \rangle$ to $\langle \mathcal{S}_\omega f, \mathcal{S}_\omega g \rangle$.

To see that Γ_ω is well-defined on arrows, let $\langle f, g \rangle$ be as above, let $\mu \in \mathcal{S}_\omega A$, $\nu \in \mathcal{S}_\omega B$ be such that $\mu\,(\Gamma_\omega R)\,\nu$, and let $U \subseteq C$, $V \subseteq D$ be such that $(\pi_1^S)^{-1}(U) \supseteq (\pi_2^S)^{-1}(V)$. An easy calculation shows that $(\pi_1^R)^{-1}(f^{-1}(U)) \supseteq (\pi_2^R)^{-1}(g^{-1}(V))$. This, together with $\mu\,(\Gamma_\omega R)\,\nu$ now gives $(\mathcal{S}_\omega f)(\mu)[U] \geq (\mathcal{S}_\omega g)(\nu)[V]$. Thus, $(\mathcal{S}_\omega f)(\mu)\,(\Gamma_\omega S)\,(\mathcal{S}_\omega g)(\nu)$.

Proposition 3.7 *Γ_ω is a relator.*

Proof. The first two requirements in the definition of a relator (see e.g. (i) and (ii) of Proposition 3.1) are immediately verified. To see that Γ_ω preserves cartesian maps, let $\langle f, g \rangle : \langle A, B, R \rangle \to \langle C, D, S \rangle$ be a relation-reflecting map, let $\mu \in \mathcal{S}_\omega A$, $\nu \in \mathcal{S}_\omega B$ be such that $(\mathcal{S}_\omega f)(\mu)\,(\Gamma_\omega S)\,(\mathcal{S}_\omega g)(\nu)$, and let $X \subseteq A$, $Y \subseteq B$ be such that $(\pi_1^R)^{-1}(X) \supseteq (\pi_2^R)^{-1}(Y)$. Also, let $U = \{ c \in C \mid c = f(a) \text{implies } a \in X \}$ and $V = g(Y)$. Then, $X \supseteq f^{-1}(U)$, $g^{-1}(V) \supseteq Y$, and $(\pi_1^S)^{-1}(U) \supseteq (\pi_2^S)^{-1}(V)$. $(\mathcal{S}_\omega f)(\mu)\,(\Gamma_\omega S)\,(\mathcal{S}_\omega g)(\nu)$ now gives $(\mathcal{S}_\omega f)(\mu)[U] \geq (\mathcal{S}_\omega g)(\nu)[V]$, and therefore $\mu[X] \geq \mu[f^{-1}(U)] \geq \nu[g^{-1}(V)] \geq \nu[Y]$. We have thus proved that $\mu\,(\Gamma_\omega R)\,\nu$. □

We now characterize the restriction of Γ_ω to Preord.

Proposition 3.8 *Let R be a preorder on A, and let $\mu, \nu \in \mathcal{S}_\omega A$. Then:*

$$\mu\,(\Gamma_\omega R)\,\nu \quad \textit{iff} \quad \mu[Y] \geq \nu[Y] \textit{ for any } R^{\mathsf{op}}\textit{-closed } Y \subseteq A \tag{5}$$

Proof. We begin by noting that, if $X, Y \subseteq A$, then $(\pi_1^R)^{-1}(X) \supseteq (\pi_2^R)^{-1}(Y)$ translates to $X \supseteq \overline{Y}$, where $\overline{Y} = \{\, a \in A \mid \exists\, y \in Y.\, a\, R\, y \,\}$. Also, the reflexivity and transitivity of R^{op} give $\overline{Y} \supseteq Y$ and $\overline{Y}$ R^{op}-closed [7]. First, let $Y \subseteq A$ be an R^{op}-closed set. Then, $(\pi_1^R)^{-1}(Y) \supseteq (\pi_2^R)^{-1}(Y)$ (as $Y \supseteq \overline{Y}$), and hence, by the definition of Γ_ω, $\mu[Y] \geq \nu[Y]$. Next, let $X, Y \subseteq A$ be such that $X \supseteq \overline{Y}$. Then, since $\overline{Y}$ is R^{op}-closed, it follows by (5) that $\mu[\overline{Y}] \geq \nu[\overline{Y}]$. We also have $\mu[X] \geq \mu[\overline{Y}]$ (as $X \supseteq \overline{Y}$) and $\nu[\overline{Y}] \geq \nu[Y]$ (as $\overline{Y} \supseteq Y$). Hence, $\mu[X] \geq \nu[Y]$. □

Next, we investigate the notion of simulation induced by Γ_ω. For simplicity, we consider Γ_ω-simulation on a single $\mathcal{S}_\omega$-coalgebra $\langle C, \gamma \rangle$. In this case, a relation $R \subseteq C \times C$ is a Γ_ω-simulation if, whenever $c\, R\, d$ and $X \subseteq C$ is R^{op}-closed, we have $\gamma(c)[X] \geq \gamma(d)[X]$. The condition that X is R^{op}-closed amounts to X being closed under simulation, that is, if $x \in X$ and y simulates x, then also $y \in X$. The requirement $\gamma(c)[X] \geq \gamma(d)[X]$ asks that a one-step transition from c is at least as likely to end in a state in X as a one-step transition from d is, whenever X is closed under simulation.

The restriction of Γ_ω to Preord satisfies the hypotheses of Proposition 2.7.

Proposition 3.9 $\Gamma_\omega : \mathsf{Preord} \to \mathsf{Preord}$ *preserves monics and is ω-accessible.*

Proof (Sketch). The key observation for proving ω-accessibility is that, for $\mu, \nu \in \mathcal{S}_\omega A$, we have:

$$\mu\,(\Gamma_\omega R)\,\nu \quad \text{iff} \quad \mu\restriction_Z\,(\Gamma_\omega(R\restriction_{Z\times Z}))\,\nu\restriction_Z$$

where $Z = \mathsf{supp}(\mu) \cup \mathsf{supp}(\nu)$, and $\mu\restriction_Z, \nu\restriction_Z \in \mathcal{S}_\omega(A \cap Z)$. □

Remark 3.10 A notion of simulation for probabilistic transition systems has also been defined in [7], namely as a preorder R on the set S of states of a probabilistic transition system, such that $s\, R\, t$ implies $\tau_a(s, X) \leq \tau_a(t, X)$ for any R-closed $X \subseteq S$ (with $\tau_a(s, X)$ giving the probability of reaching a state in X via an a-labelled transition from s). It then follows by the previous characterization of $\Gamma_\omega : \mathsf{Preord} \to \mathsf{Preord}$ that R is a simulation preorder according to [7] (in the unlabelled case) if and only if R^{op} is a simulation preorder w.r.t. Γ_ω.

[7] Given a preorder $\langle A, R \rangle$, a subset $Y \subseteq A$ is R^{op}*-closed* if $y \in Y$ and $a\, R\, y$ imply $a \in Y$.

4 Logics for Simulation

We now describe an inductive method for defining logics which capture simulation. We use a notion of language constructor to capture one step in the definition of a language for coalgebras, and show that the expressiveness of the resulting language w.r.t. a given notion of simulation follows from an expressivity condition involving the language constructor and the given relator.

4.1 Basic Definitions

The notion of language, as defined below, will be needed when defining language constructors. A variant of this notion was used in [5].

Definition 4.1 A **language** is a tuple $\langle X, \mathcal{L}, \models \rangle$, with X a set (the semantic domain), $\mathcal{L}$ a set (of formulae) containing a distinguished element $\top$, and $\models \subseteq X \times \mathcal{L}$ a binary relation such that $X \times \{\top\} \subseteq \models$.

A **map between languages** $\langle X, \mathcal{L}, \models \rangle$ and $\langle X', \mathcal{L}', \models' \rangle$ is a pair $\langle f, l \rangle$, with $f : X' \to X$ and $l : \mathcal{L} \to \mathcal{L}'$ being such that:

(i) $l(\top) = \top$,

(ii) $f(x') \models \varphi$ if and only if $x' \models' l(\varphi)$, for $x' \in X'$ and $\varphi \in \mathcal{L}$.

The category of languages and maps between them is denoted Lang.

Thus, the only propositional structure which is required of a language is the formula $\top$, interpreted as true. Additional propositional structure, including conjunction and disjunction, will be required in concrete examples.

Given a language $\langle X, \mathcal{L}, \models \rangle$ and a formula $\varphi \in \mathcal{L}$, we write $[\![\varphi]\!]$ for the set $\{ x \in X \mid x \models \varphi \}$.

Remark 4.2 Any language $\langle X, \mathcal{L}, \models \rangle$ induces a *logical map* $s : X \to \mathcal{P}\mathcal{L}$, defined by $s(x) = \{ \varphi \in \mathcal{L} \mid x \models \varphi \}$ for $x \in X$. Then, condition (ii) defining maps between languages is equivalent to $s \circ f = \hat{\mathcal{P}} l \circ s'$, where $\hat{\mathcal{P}} : \mathsf{Set} \to \mathsf{Set}$ is the contravariant powerset functor.

We let $\mathsf{E} : \mathsf{Lang} \to \mathsf{Set}^{op}$ denote the functor taking languages to their semantic domains, and maps between languages to the underlying functions between the semantic domains. The next two results have been proved in [5] for a slightly different notion of language, but they also hold in the present setting.

Lemma 4.3 E *is a cofibration*[8] *.*

[8] See [3] for a definition.

Proof (Sketch). Given $\langle X, \mathcal{L}, \models \rangle$ and $f : X' \to X$, let $\models' \subseteq X' \times \mathcal{L}$ be given by $x' \models' \varphi$ if and only if $f(x') \models \varphi$. Then, $\langle f, 1_{\mathcal{L}} \rangle : \langle X, \mathcal{L}, \models \rangle \to \langle X', \mathcal{L}, \models' \rangle$ is a cocartesian map. □

Proposition 4.4 Lang *is cocomplete.*

Proof (Sketch). Colimits in Lang are constructed from colimits in $\mathsf{Set}^{\mathsf{op}}$ and colimits in certain fibres of E. □

For instance, an initial object in Lang is given by the language $\langle 1, \{\top\}, \models \rangle$ with $\models = 1 \times \{\top\}$. Proposition 4.4 will later allows us to join languages with different (but related) semantic domains.

We now use the notion of language constructor (a variant of which was introduced in [5]) to formalise one step in the definition of a language for coalgebras.

Definition 4.5 Let $\mathsf{T} : \mathsf{Set} \to \mathsf{Set}$ be an arbitrary endofunctor. A **language constructor for** T is an endofunctor $\mathcal{F} : \mathsf{Lang} \to \mathsf{Lang}$ satisfying $\mathsf{E} \circ \mathcal{F} = \mathsf{T}^{\mathsf{op}} \circ \mathsf{E}$.

Thus, a language constructor takes a language of form $\langle X, \mathcal{L}, \models \rangle$ to a language of form $\langle \mathsf{T}X, \mathcal{L}', \models' \rangle$.

Example 4.6 Let $\mathcal{F}_{\supseteq} : \mathsf{Lang} \to \mathsf{Lang}$ denote the language constructor for $\mathcal{P}_\omega$ which takes $\langle X, \mathcal{L}, \models \rangle$ to $\langle \mathcal{P}_\omega X, (\{\top\} \cup \{ \Diamond\varphi \mid \varphi \in \mathcal{L} \})^\wedge, \models' \rangle$, where $(\_)^\wedge$ denotes closure under binary conjunctions, and where $\models'$ is the natural extension of the relation defined by:

$$Y \models' \Diamond\varphi \text{ iff } \exists\, y \in Y.\, y \models \varphi \qquad (Y \in \mathcal{P}_\omega X \,,\, \varphi \in \mathcal{L})$$

to formulae containing conjunctions.

In [5], we were interested in the ability of (the formulae of) a language to characterize elements of the underlying semantic domain. Here, we are interested in characterizing certain relations on the semantic domain.

Definition 4.7 Let $\langle X, \mathcal{L}, \models \rangle$ be a language, and let $\langle X, R \rangle$ be a preorder. Given $x, y \in X$, we write $y \geq_{\mathcal{L}} x$ if $y \models \varphi$ whenever $x \models \varphi$, with $\varphi \in \mathcal{L}$. Then, $\langle X, \mathcal{L}, \models \rangle$ is called **adequate for** $\langle X, R \rangle$ if $R \subseteq \geq_{\mathcal{L}}$, and **expressive for** $\langle X, R \rangle$ if, in addition, $R \supseteq \geq_{\mathcal{L}}$.

Thus, adequacy of a language $\langle X, \mathcal{L}, \models \rangle$ for a preorder $\langle X, R \rangle$ amounts to the logical map $s : X \to \mathcal{P}\mathcal{L}$ defining a map $s : \langle X, R \rangle \to \langle \mathcal{P}\mathcal{L}, \supseteq \rangle$ in Preord, whereas expressiveness of $\langle X, \mathcal{L}, \models \rangle$ for $\langle X, R \rangle$ amounts to s being a cartesian (or order-reflecting) map.

Definition 4.8 Let $\Gamma : \mathsf{Rel} \to \mathsf{Rel}$ be a T-relator. A language constructor for T **preserves expressiveness w.r.t.** Γ if it takes a language $\langle X, \mathcal{L}, \models \rangle$ expressive for $\langle X, R\rangle$ to a language $\langle \mathsf{T}X, \mathcal{L}', \models' \rangle$ expressive for $\langle \mathsf{T}X, \Gamma R\rangle$.

Example 4.9 It is relatively easy to check that the language constructor $\mathcal{F}_{\supseteq}$ from Example 4.6 preserves expressiveness w.r.t. $\Gamma_{\supseteq}$. The only challenge is to define a formula $\phi \in \mathcal{L}'$ which holds in Z but not in Y, whenever $Y\,(\Gamma_{\supseteq} R)\,Z$ does not hold (having assumed that $\mathcal{L}$ is expressive w.r.t. R). First, the fact that $Y\,(\Gamma_{\supseteq} R)\,Z$ does not hold gives $z \in Z$ such that $y\,R\,z$ does not hold for any $y \in Y$. The expressiveness of $\mathcal{L}$ w.r.t. R then gives, for each $y \in Y$, a formula φ_y such that $z \models \varphi_y$ but $y \not\models \varphi_y$. Then, the formula $\Diamond(\bigwedge_{y\in Y} \varphi_y)$ holds in Z but not in Y.

The next two subsections contain two more examples of language constructors. We first consider a language constructor which mirrors the construction of Baltag's logics for coalgebraic simulation [1], and prove that it preserves expressiveness w.r.t. the underlying relator. We then define a language constructor for probabilistic transition systems, and prove that it preserves expressiveness w.r.t. the relator defined in Section 3.1.

4.2 Baltag's Logics for Coalgebraic Simulation

Let $\mathsf{T} : \mathsf{Set} \to \mathsf{Set}$ be a functor which preserves inclusions, and let $\Gamma : \mathsf{Rel} \to \mathsf{Rel}$ be a T-relator. Also, let $\mathcal{F}_\Gamma : \mathsf{Lang} \to \mathsf{Lang}$ be defined by:

- $\mathcal{F}_\Gamma$ takes $\langle X, \mathcal{L}, \models\rangle$ to $\langle \mathsf{T}X, (\mathsf{T}\mathcal{L})^\wedge, (\Gamma{\models})^\wedge\rangle$, where $(\mathsf{T}\mathcal{L})^\wedge$ denotes the closure of $\mathsf{T}\mathcal{L}$ under arbitrary conjunctions, and $(\Gamma{\models})^\wedge$ is the natural extension of $\Gamma{\models}$ to formulae containing conjunctions. We let $\top = \bigwedge \emptyset \in (\mathsf{T}\mathcal{L})^\wedge$.
- $\mathcal{F}_\Gamma$ takes $\langle f, l\rangle : \langle X_1, \mathcal{L}_1, \models_1\rangle \to \langle X_2, \mathcal{L}_2, \models_2\rangle$ to $\langle \mathsf{T}f, (\mathsf{T}l)^\wedge\rangle$, where $(\mathsf{T}l)^\wedge : (\mathsf{T}\mathcal{L}_1)^\wedge \to (\mathsf{T}\mathcal{L}_2)^\wedge$ denotes the unique extension of $\mathsf{T}l$ to a function preserving conjunctions.

For $\mathcal{F}_\Gamma$ to be well-defined, we must prove that:

$$(\mathsf{T}f)(t)\,(\Gamma{\models_1})\,\phi \quad \text{iff} \quad t\,(\Gamma{\models_2})\,(\mathsf{T}l)(\phi) \tag{6}$$

for any $\langle f, l\rangle : \langle X_1, \mathcal{L}_1, \models_1\rangle \to \langle X_2, \mathcal{L}_2, \models_2\rangle$, any $t \in \mathsf{T}X_2$ and any $\phi \in (\mathsf{T}\mathcal{L}_1)^\wedge$.

Lemma 4.10 *Let $\langle X, \mathcal{L}, \models\rangle$ be a language with logical map $s : X \to \mathcal{P}\mathcal{L}$, and let $e : \mathsf{T}\hat{\mathcal{P}} \Rightarrow \hat{\mathcal{P}}\mathsf{T}$ be given by $e_X(U) = \{\, t \in \mathsf{T}X \mid U\,(\Gamma{\ni})\,t \,\}$ for $X \in |\mathsf{Set}|$ and $U \in \mathsf{T}\mathcal{P}X$* [9]*. Then:*

[9] Here, $\ni$ denotes the converse of the membership relation.

(i) *e is a natural transformation;*

(ii) *The logical map $s' : \mathsf{T}X \to \mathcal{P}\mathsf{T}\mathcal{L}$ induced by $(\Gamma\models) \subseteq \mathsf{T}X \times \mathsf{T}\mathcal{L}$ is given by $e_{\mathcal{L}} \circ \mathsf{T}s$.*

Proof. We note that, for $f : X \to Y$, $\ni \circ \mathsf{Gr}(\hat{\mathcal{P}}f) = \mathsf{Gr}(f)^{\mathsf{op}} \circ \ni$. Preservation of relational composition by Γ together with Remark 3.3 then give $(\Gamma\ni) \circ \mathsf{Gr}(\mathsf{T}\hat{\mathcal{P}}f) = \mathsf{Gr}(\mathsf{T}f)^{\mathsf{op}} \circ (\Gamma\ni)$, i.e. $(\mathsf{T}\hat{\mathcal{P}}f)(V)\,(\Gamma\ni)\,t$ if and only if $V\,(\Gamma\ni)\,(\mathsf{T}f)(t)$, for $V \in \mathsf{T}\hat{\mathcal{P}}Y$ and $t \in \mathsf{T}X$. But this is equivalent to $e_X \circ \mathsf{T}\hat{\mathcal{P}}f = \hat{\mathcal{P}}\mathsf{T}f \circ e_Y$. Hence, e is natural.

We also note that the definition of s makes $\langle s, 1_{\mathcal{L}}\rangle : \langle X, \mathcal{L}, \models\rangle \to \langle \mathcal{P}\mathcal{L}, \mathcal{L}, \ni\rangle$ a cartesian map. Preservation of cartesian maps by Γ then gives $t\,(\Gamma\models)\,\phi$ if and only $(\mathsf{T}s)(t)\,(\Gamma\ni)\,\phi$, for $t \in \mathsf{T}X$ and $\phi \in \mathsf{T}\mathcal{L}$. That is, $\phi \in s'(t)$ if and only if $\phi \in e_{\mathcal{L}}((\mathsf{T}s)(t))$. Hence, $s' = e_{\mathcal{L}} \circ \mathsf{T}s$. □

We now return to proving (6). Here we only consider the case when $\phi \in \mathsf{T}\mathcal{L}$. (The remaining case follows by induction.) In this case, (6) is equivalent to $s'_1 \circ \mathsf{T}f = \hat{\mathcal{P}}\mathsf{T}l \circ s'_2$ (see Remark 4.2), where $s'_1 : \mathsf{T}X_1 \to \mathcal{P}\mathsf{T}\mathcal{L}_1$ and $s'_2 : \mathsf{T}X_2 \to \mathcal{P}\mathsf{T}\mathcal{L}_2$ are the logical maps induced by $\Gamma\models_1$ and $\Gamma\models_2$. By (ii) of Lemma 4.10, this is equivalent to $e_{\mathcal{L}_1} \circ \mathsf{T}s_1 \circ \mathsf{T}f = \hat{\mathcal{P}}\mathsf{T}l \circ e_{\mathcal{L}_2} \circ \mathsf{T}s_2$, which, in turn, is a consequence of (i) of Lemma 4.10 and of Remark 4.2. Thus, $\mathcal{F}_\Gamma$ is well-defined.

Proposition 4.11 *$\mathcal{F}_\Gamma$ preserves expressiveness w.r.t. Γ.*

Proof. We begin by showing that, if $\langle X, \mathcal{L}, \models\rangle$ is adequate for $\langle X, R\rangle$, then $\langle \mathsf{T}X, \mathsf{T}\mathcal{L}, \Gamma\models\rangle$ is adequate for $\langle \mathsf{T}X, \Gamma R\rangle$ (and hence so is $\mathcal{F}_\Gamma\langle X, \mathcal{L}, \models\rangle$). The adequacy of $\langle X, \mathcal{L}, \models\rangle$ for $\langle X, R\rangle$ translates to $\models \circ R \subseteq \models$. The preservation of inclusions by T and of relational composition by Γ then give $(\Gamma\models) \circ (\Gamma R) \subseteq (\Gamma\models)$. That is, $\langle \mathsf{T}X, \mathsf{T}\mathcal{L}, \Gamma\models\rangle$ is adequate for $\langle \mathsf{T}X, \Gamma R\rangle$.

Now assume that $\langle X, \mathcal{L}, \models\rangle$ is expressive for $\langle X, R\rangle$, i.e. $R = \geq_{\mathcal{L}}$. Following [1], we define $\theta : X \to \mathcal{L}$ by $\theta(x) = \bigwedge_{\varphi \in \mathcal{L}, x \models \varphi} \varphi$. Then:

$$y\,R\,x \;\text{ iff }\; y \geq_{\mathcal{L}} x \;\text{ iff }\; y \models \theta(x) \;\text{ iff }\; y\,(\mathsf{Gr}(\theta)^{\mathsf{op}} \circ \models)\,x \tag{7}$$

The definition of θ also gives $\mathsf{Gr}(\theta) \subseteq \models$, and hence:

$$\mathsf{Gr}(\mathsf{T}\theta) \;\subseteq\; (\Gamma{=_{\mathcal{L}}}) \circ \mathsf{Gr}(\mathsf{T}\theta) \;=\; \Gamma(\mathsf{Gr}(\theta)) \;\subseteq\; (\Gamma\models) \tag{8}$$

The first inclusion follows from the definition of a relator (alternatively see (ii) of Proposition 3.1), the subsequent equality follows by (4), and the final inclusion follows from the preservation of inclusions by T and Γ. We then have:

$$\Gamma R \;=\; (\Gamma\mathsf{Gr}(\theta)^{\mathsf{op}}) \circ (\Gamma\models) \;=\; \mathsf{Gr}(\mathsf{T}\theta)^{\mathsf{op}} \circ (\Gamma\models) \;\supseteq\; \geq_{\mathsf{T}\mathcal{L}}$$

The first equality follows from (7) using the preservation of relational composition by Γ, while the second equality follows by Remark 3.3. To prove the containment relation, let $v \geq_{\mathsf{T}\mathcal{L}} u$. Then, $u\,(\Gamma \models)\,(\mathsf{T}\theta)(u)$ (by (8)), and hence $v\,(\Gamma\models)\,(\mathsf{T}\theta)(u)$. This, together with $(\mathsf{T}\theta)(u)\,\mathsf{Gr}(\mathsf{T}\theta)^{\mathsf{op}}\,u$ now yields $v\,(\mathsf{Gr}(\mathsf{T}\theta)^{\mathsf{op}} \circ (\Gamma \models))\,u$. We have therefore proved that $\Gamma R \supseteq \geq_{\mathsf{T}\mathcal{L}}$. Hence, $\langle \mathsf{T}X, \mathsf{T}\mathcal{L}, \Gamma\models\rangle$ is expressive for $\langle \mathsf{T}X, \Gamma R\rangle$. □

Thus, for an inclusion-preserving endofunctor $\mathsf{T} : \mathsf{Set} \to \mathsf{Set}$ and a T-relator $\Gamma : \mathsf{Rel} \to \mathsf{Rel}$, the language constructor $\mathcal{F}_\Gamma$ formalises one step in the definition of a language for T-coalgebras.

4.3 Probabilistic Transition Systems

Let $\Gamma_\omega : \mathsf{Rel} \to \mathsf{Rel}$ be as in Section 3.1, and define $\mathcal{F}_\omega : \mathsf{Lang} \to \mathsf{Lang}$ by:

- $\mathcal{F}_\omega$ takes $\langle X, \mathcal{L}, \models\rangle$ to $\langle \mathcal{S}_\omega X, \mathcal{L}', \models'\rangle$, where $\mathcal{L}' = (\{\top\} \cup \{\, \diamond_p\varphi \mid p \in \mathbb{Q} \cap [0,1]\,,\, \varphi \in \mathcal{L}\,\})^{\wedge,\vee}$ (with $(\_)^{\wedge,\vee}$ denoting closure under binary conjunctions and disjunctions), and where $\models'$ is the natural extension of the relation defined by:
$$\mu \models' \diamond_p\varphi \;\text{ iff }\; \mu[\![\varphi]\!] \geq p$$
to formulae containing conjunctions and disjunctions.
- $\mathcal{F}_\omega$ takes $\langle f, l\rangle : \langle X_1, \mathcal{L}_1, \models_1\rangle \to \langle X_2, \mathcal{L}_2, \models_2\rangle$ to $\langle \mathcal{S}_\omega f, l'\rangle$, where $l' : \mathcal{L}_1' \to \mathcal{L}_2'$ takes $\diamond_p\varphi$ to $\diamond_p l(\varphi)$ and distributes over conjunctions and disjunctions.

Thus, a formula of form $\diamond_p\varphi$ holds for a finite sub-probability distribution μ if a state satisfying φ is reached via μ with probability at least p.

Remark 4.2 can be used to show that $\mathcal{F}_\omega$ is well-defined on arrows.

Proposition 4.12 *$\mathcal{F}_\omega$ preserves expressiveness w.r.t. Γ_ω.*

Proof. First, assume $\langle X, \mathcal{L}, \models\rangle$ is adequate for $\langle X, R\rangle$. We immediately infer that $[\![\varphi]\!]$ is R^{op}-closed for any $\varphi \in \mathcal{L}$. To show that $\langle \mathcal{S}_\omega X, \mathcal{L}', \models'\rangle$ is adequate for $\langle \mathcal{S}_\omega X, \Gamma_\omega R\rangle$, let $\mu, \nu \in \mathcal{S}_\omega X$ be such that $\mu\,(\Gamma_\omega R)\,\nu$. The proof of the fact that $\nu \models \phi$ implies $\mu \models \phi$ for all $\phi \in \mathcal{L}'$ (and hence $\mu \geq_{\mathcal{L}'} \nu$) is by induction on ϕ. The non-trivial case is when ϕ is of form $\diamond_p\varphi$ with $\varphi \in \mathcal{L}$. In this case, $\nu \models \phi$ translates to $\nu[\![\varphi]\!] \geq p$. Also, since $[\![\varphi]\!]$ is R^{op}-closed, it follows that $\mu[\![\varphi]\!] \geq \nu[\![\varphi]\!]$. Hence, $\mu[\![\varphi]\!] \geq p$, that is, $\mu \models \phi$.

Now assume $\langle X, \mathcal{L}, \models\rangle$ is expressive for $\langle X, R\rangle$. To show that $\langle \mathcal{S}_\omega X, \mathcal{L}', \models'\rangle$ is expressive for $\langle \mathcal{S}_\omega X, \Gamma_\omega R\rangle$, we must prove that $\mu[Y] \geq \nu[Y]$ for any R^{op}-closed $Y \subseteq X$, whenever $\mu, \nu \in \mathcal{S}_\omega X$ are such that $\mu \geq_{\mathcal{L}'} \nu$. We can assume that $Y \neq \emptyset$ (otherwise $\mu[Y] = \nu[Y] = 0$ and we are done). We note that, for any R^{op}-closed $\emptyset \neq Y \subseteq X$, $Y = \bigcup\limits_{y \in Y} \bigcap\limits_{y \models \varphi} [\![\varphi]\!]$: the left-to-right inclusion is immediate, whereas the right-to-left inclusion follows from the expressiveness of $\langle X, \mathcal{L}, \models\rangle$ for $\langle X, R\rangle$ together with Y being R^{op}-closed. Thus, if both Y and the sets $\{\varphi \mid \varphi \in \mathcal{L},\ y \models \varphi\}$ with $y \in Y$ are finite, the formulae $\Diamond_p \varphi_Y$, with $p \in \mathbb{Q} \cap [0, \nu[Y]]$ and $\varphi_Y = \bigvee_{y \in Y} \bigwedge_{y \models \varphi} \varphi$ can be used to show that $\mu[Y] \geq \nu[Y]$. For, $\nu \models \Diamond_p \varphi_Y$ yields $\mu \models \Diamond_p \varphi_Y$ for any $p \in \mathbb{Q} \cap [0, \nu[Y]]$. That is, $\mu[Y] = \mu[[\![\varphi_Y]\!]] \geq p$ for any $p \in \mathbb{Q} \cap [0, \nu[Y]]$. This, in turn, gives $\mu[Y] \geq \nu[Y]$.

However, the previously-mentioned sets are not, in general, finite. Nevertheless, it is possible to define a formula $\varphi \in \mathcal{L}$ with the property that $\mu[Y] = \mu[[\![\varphi]\!]]$ and $\nu[Y] = \nu[[\![\varphi]\!]]$. Then, the above reasoning can be applied to the formulae $\Diamond_p \varphi$ with $p \in \mathbb{Q} \cap [0, \nu[Y]]$. In order to define φ, let $Z = \mathsf{supp}(\mu) \cup \mathsf{supp}(\nu)$, and let $\equiv$ denote the equivalence relation on $\mathcal{L}$ given by $\varphi_1 \equiv \varphi_2$ if and only if $[\![\varphi_1]\!] \cap Z = [\![\varphi_2]\!] \cap Z$. Since Z is finite, there are only finitely-many equivalence classes w.r.t. $\equiv$. For $y \in Y$, let $\Phi_y = \{\varphi \in \mathcal{L} \mid y \models \varphi\}$, and let $\Phi_y^0 \subseteq \Phi_y$ consist of a set of representatives for Φ_y. Then, for $z \in Z$, $z \models \varphi$ for all $\varphi \in \Phi_y$ if and only $z \models \varphi$ for all $\varphi \in \Phi_y^0$. Now let $\Phi = \{\bigwedge_{\varphi \in \Phi_y^0} \varphi \mid y \in Y\}$, and let $\Phi^0 \subseteq \Phi$ consists of a set of representatives for Φ. Then, for $z \in Z$, $z \models \phi$ for some $\phi \in \Phi$ if and only if $z \models \phi$ for some $\phi \in \Phi^0$. One can therefore infer that, for $z \in Z$, $z \in Y$ if and only if $z \models \bigvee_{\phi \in \Phi^0} \phi$. This, in turn, gives $\mu[Y] = \mu[[\![\bigvee_{\phi \in \Phi^0} \phi]\!]]$ and $\nu[Y] = \nu[[\![\bigvee_{\phi \in \Phi^0} \phi]\!]]$. Then, $\mu \geq_{\mathcal{L}'} \nu$ together with $\nu \models \Diamond_p \bigvee_{\phi \in \Phi^0} \phi$ gives $\mu \models \Diamond_p \bigvee_{\phi \in \Phi^0} \phi$, or equivalently $\mu[Y] \geq p$, for all $p \in \mathbb{Q} \cap [0, \nu[Y]]$. Hence, $\mu[Y] \geq \nu[Y]$. □

4.4 Logics for Coalgebraic Simulation

We now fix an endofunctor $\mathsf{T} : \mathsf{Set} \to \mathsf{Set}$ and a T-relator $\Gamma : \mathsf{Rel} \to \mathsf{Rel}$, and let $\gtrsim = \gtrsim_\Gamma$ denote the similarity relation induced by Γ. We are interested in languages for T-coalgebras which capture Γ-similarity.

Definition 4.13 A **language for T-coalgebras** is a pair $\langle \mathcal{L}, \models\rangle$ with $\mathcal{L}$ a set and $\models = (\models_\gamma)$ a $|\mathsf{Coalg}(\mathsf{T})|$-indexed family of relations $\models_\gamma \subseteq C \times \mathcal{L}$ for $\gamma : C \to \mathsf{T}C$, such that:

$$f(c) \models_\delta \varphi \ \text{ iff } \ c \models_\gamma \varphi \quad \text{for any } f : \langle C, \gamma\rangle \to \langle D, \delta\rangle\,,\ c \in C\,,\ \varphi \in \mathcal{L}\,.$$

Given $\langle \mathcal{L}, \models\rangle$ and T-coalgebras $\langle C, \gamma\rangle$ and $\langle D, \delta\rangle$, we say that $c \in C$ **logically simulates** $d \in D$ (and write $c \geq_{\mathcal{L}} d$) if $c \models_\gamma \varphi$ whenever $d \models_\delta \varphi$, for any $\varphi \in \mathcal{L}$. $\langle \mathcal{L}, \models\rangle$ is said to **capture Γ-similarity** if, for any T-coalgebras $\langle C, \gamma\rangle$ and $\langle D, \delta\rangle$, the logical simulation relation $\geq_{\mathcal{L}} \subseteq C \times D$ coincides with the

Γ-similarity relation $\gtrsim \subseteq C \times D$.

Remark 4.14 Let α be a regular cardinal. Then, any language of form $\langle Z_\alpha, \mathcal{L}_\alpha, \models_\alpha \rangle$ induces a language $\langle \mathcal{L}_\alpha, \models \rangle$ for T-coalgebras, with $c \models_\gamma \varphi$ if and only if $\gamma_\alpha(c) \models_\alpha \varphi$, for any T-coalgebra $\langle C, \gamma \rangle$, $c \in C$ and $\varphi \in \mathcal{L}$ (where $\gamma_\alpha : C \to Z_\alpha$ is as in Remark 2.6). The fact that coalgebra homomorphisms $f : \langle C, \gamma \rangle \to \langle D, \delta \rangle$ define morphisms of cones $f : (\gamma_\alpha) \to (\delta_\alpha)$ ensures the correctness of this definition.

Now assume that T admits a final coalgebra $\langle F, \zeta \rangle$, and recall from Remark 2.15 that, if c and d are as in Definition 4.13, then $c \gtrsim d$ if and only if $!_\gamma(c) \gtrsim !_\delta(d)$. Also, Definition 4.13 gives $c \geq_\mathcal{L} d$ if and only if $!_\gamma(c) \geq_\mathcal{L} !_\delta(d)$. Thus, in order to define a language for T-coalgebras which captures Γ-similarity, it suffices to define a language of form $\langle F, \mathcal{L}, \models \rangle$ which is expressive for $\langle F, \gtrsim \rangle$. But by (iii) of Proposition 2.14, $\langle \langle F, F, \gtrsim \rangle, \langle \zeta, \zeta \rangle \rangle$ is a final Γ-coalgebra. This leads us to consider the final sequence of $\Gamma : \mathsf{Rel} \to \mathsf{Rel}$.

Proposition 4.15 *The final sequence of* Γ *belongs to* Preord.

Proof (Sketch). The statement follows by transfinite induction. Proposition 2.2 is used in the case of limit ordinals. □

As a result, the final sequence of Γ coincides with the final sequence of the restriction of Γ to Preord. This justifies the following definition.

Definition 4.16 The **relation sequence induced by** Γ is the final sequence of $\Gamma : \mathsf{Preord} \to \mathsf{Preord}$.

An immediate observation is that the Set-sequence underlying the relation sequence induced by Γ is the final sequence of T. Thus, the relation sequence induced by Γ can be written $(\langle Z_\alpha, \gtrsim_\alpha \rangle)$, $(p^\alpha_\beta : \langle Z_\alpha, \gtrsim_\alpha \rangle \to \langle Z_\beta, \gtrsim_\beta \rangle)_{\beta \leq \alpha}$.

The next step is to define, for each element Z_α in the final sequence of T, an expressive language for $\langle Z_\alpha, \gtrsim_\alpha \rangle$. A similar definition was given in [5].

Definition 4.17 Let $\mathcal{F} : \mathsf{Lang} \to \mathsf{Lang}$ be a language constructor for T. The **language sequence induced by** $\mathcal{F}$ is the initial sequence [10] of $\mathcal{F}$.

Again, the Set-sequence underlying the language sequence induced by $\mathcal{F}$ is the final sequence of T. We therefore write $(\langle Z_\alpha, \mathcal{L}_\alpha, \models_\alpha \rangle)$, $(\langle p^\alpha_\beta, \iota^\alpha_\beta \rangle : \langle Z_\beta, \mathcal{L}_\beta, \models_\beta \rangle \to \langle Z_\alpha, \mathcal{L}_\alpha, \models_\alpha \rangle)_{\beta \leq \alpha}$ for the language sequence induced by $\mathcal{F}$, and $s_\alpha : Z_\alpha \to \mathcal{P}\mathcal{L}_\alpha$ for the logical map induced by $\langle Z_\alpha, \mathcal{L}_\alpha, \models_\alpha \rangle$.

The next result concerns the expressiveness of languages in the language sequence induced by $\mathcal{F}$ w.r.t. relations in the relation sequence induced by Γ.

[10] The initial sequence of an endofunctor is defined similarly to its final sequence.

Theorem 4.18 *If $\mathcal{F}$ preserves expressiveness w.r.t. Γ, then $\langle Z_\alpha, \mathcal{L}_\alpha, \models_\alpha \rangle$ is expressive for $\langle Z_\alpha, \gtrsim_\alpha \rangle$, for any ordinal α.*

Proof. The proof is by induction on α. If $\alpha = \beta + 1$, the expressiveness of $\langle Z_\alpha, \mathcal{L}_\alpha, \models_\alpha \rangle$ for $\langle Z_\alpha, \gtrsim_\alpha \rangle$ follows from the expressiveness of $\langle Z_\beta, \mathcal{L}_\beta, \models_\beta \rangle$ for $\langle Z_\beta, \gtrsim_\beta \rangle$ together with the preservation of expressiveness by $\mathcal{F}$.

Now let α be a limit ordinal, and assume that $\langle Z_\beta, \mathcal{L}_\beta, \models_\beta \rangle$ is expressive for $\langle Z_\beta, \gtrsim_\beta \rangle$, for any $\beta < \alpha$. To show that $\langle Z_\alpha, \mathcal{L}_\alpha, \models_\alpha \rangle$ is adequate for $\langle Z_\alpha, \gtrsim_\alpha \rangle$, let $x, y \in Z_\alpha$ be such that $y \gtrsim_\alpha x$. Then, for $\beta < \alpha$, $p^\alpha_\beta(y) \gtrsim_\beta p^\alpha_\beta(x)$, and hence, using the adequacy of $\langle Z_\beta, \mathcal{L}_\beta, \models_\beta \rangle$ for $\langle Z_\beta, \gtrsim_\beta \rangle$, $s_\beta(p^\alpha_\beta(y)) \supseteq s_\beta(p^\alpha_\beta(x))$. Then, Remark 4.2 gives $(\hat{\mathcal{P}}\iota^\alpha_\beta)(s_\alpha(y)) \supseteq (\hat{\mathcal{P}}\iota^\alpha_\beta)(s_\alpha(x))$ for $\beta < \alpha$. Now let $\varphi \in s_\alpha(x)$. Since the cocone $(\iota^\alpha_\beta)_{\beta<\alpha}$ is colimiting, we have $\varphi = \iota^\alpha_\beta(\psi)$ for some $\beta < \alpha$ and some $\psi \in \mathcal{L}_\beta$. Then, $\psi \in s_\beta(p^\alpha_\beta(x))$, and hence $\psi \in s_\beta(p^\alpha_\beta(y))$ (or equivalently, $\psi \in (\hat{\mathcal{P}}\iota^\alpha_\beta)(s_\alpha(y))$). This now gives $\varphi = \iota^\alpha_\beta(\psi) \in s_\alpha(y)$. Hence, $s_\alpha(y) \supseteq s_\alpha(x)$.

To show that $\langle Z_\alpha, \mathcal{L}_\alpha, \models_\alpha \rangle$ is expressive for $\langle Z_\alpha, \gtrsim_\alpha \rangle$, let $x, y \in Z_\alpha$ be such that $s_\alpha(y) \supseteq s_\alpha(x)$. Then, for $\beta < \alpha$, Remark 4.2 gives $s_\beta(p^\alpha_\beta(y)) \supseteq s_\beta(p^\alpha_\beta(x))$, while the expressiveness of $\langle Z_\beta, \mathcal{L}_\beta, \models_\beta \rangle$ for $\langle Z_\beta, \gtrsim_\beta \rangle$ gives $p^\alpha_\beta(y) \gtrsim_\beta p^\alpha_\beta(x)$. The fact that the cone $(p^\alpha_\beta)_{\beta<\alpha}$ is limiting now gives $y \gtrsim_\alpha x$. □

Our aim is to derive a language for T-coalgebras which captures $\gtrsim$.

Definition 4.19 Assume that the final sequence of Γ stabilizes at α. The **language induced by** $\langle \mathcal{F}, \Gamma \rangle$ is the language $\langle \mathcal{L}_\alpha, \models \rangle$, as defined in Remark 4.14[11] .

Example 4.20 Let $\Gamma_\supseteq$ and $\mathcal{F}_\supseteq$ be as in Examples 2.9 and 4.6, respectively. Since $\Gamma_\supseteq$ preserves monics and is ω-accessible, it follows by Proposition 2.7 that its final sequence stabilizes. Moreover, the initial sequence of the endofunctor $\mathsf{L} : \mathsf{Set} \to \mathsf{Set}$ taking $\mathcal{L}$ to $(\{\top\} \cup \{ \Diamond\varphi \mid \varphi \in \mathcal{L} \})^\wedge$ stabilises at ω. (L defines the syntax part of $\mathcal{F}_\supseteq$.) As a result, the language induced by $\langle \mathcal{F}_\supseteq, \Gamma_\supseteq \rangle$ coincides with the following fragment of the standard modal language:

$$\varphi ::= \top \mid \Diamond\varphi \mid \varphi \wedge \psi$$

The coalgebraic semantics of this fragment is defined by:

$$c \models_\gamma \Diamond\varphi \;\text{ iff }\; \exists\, d \in \gamma(c).\, d \models \varphi$$

(and the usual clauses for $\top$ and $\wedge$).

Example 4.21 Let $\mathcal{F}_\Gamma$ be as in Section 4.2. The language induced by $\langle \mathcal{F}_\Gamma, \Gamma \rangle$ coincides with a fragment of the language defined in [1], only containing $\top$

[11] Here, $\langle Z_\alpha, \mathcal{L}_\alpha, \models_\alpha \rangle$ is the α-indexed element of the language sequence induced by $\mathcal{F}$.

and $\bigwedge$ as propositional connectives. Its coalgebraic semantics agrees with [1].

Example 4.22 Let Γ_ω and $\mathcal{F}_\omega$ be as in Sections 3.1 and 4.3. Using an argument similar to the one in Example 4.20, we obtain that the language induced by $\langle \mathcal{F}_\omega, \Gamma_\omega \rangle$ has syntax given by:

$$\varphi ::= \top \mid \Diamond_p \varphi \mid \varphi \wedge \psi \mid \varphi \vee \psi$$

and semantics given by:

$$c \models_\gamma \Diamond_p \varphi \;\text{ iff }\; \gamma(c)[[\![\varphi]\!]_\gamma] \geq p$$

where $[\![\varphi]\!]_\gamma = \{ c \in C \mid c \models_\gamma \varphi \}$. This language coincides with the unlabelled version of the language considered in [7].

The next result allows us to derive languages which capture Γ-similarity from language constructors which preserve expressiveness w.r.t. Γ.

Corollary 4.23 *Let Γ and $\mathcal{F}$ be as in Theorem 4.18, and assume that Γ preserves monics and is accessible*[12]*. Then, the language induced by $\langle \mathcal{F}, \Gamma \rangle$ captures $\gtrsim$.*

Proof. Let $\langle C, \gamma \rangle$ and $\langle D, \delta \rangle$ be T-coalgebras, and let $c \in C$ and $d \in D$. Then:

$$c \gtrsim d \;\text{ iff }\; !_\gamma(c) \gtrsim !_\delta(d) \;\text{ iff }\; !_\gamma(c) \gtrsim_\alpha !_\delta(d) \;\text{ iff }\; !_\gamma(c) \geq_{\mathcal{L}_\alpha} !_\delta(d) \;\text{ iff }\; c \geq_{\mathcal{L}_\alpha} d$$

The above equivalences follow from Remark 2.15, (iii) of Proposition 2.14, Theorem 4.18 and Definition 4.13, respectively. □

By instantiating $\langle \mathcal{F}, \Gamma \rangle$ with $\langle \mathcal{F}_\Gamma, \Gamma \rangle$ and $\langle \mathcal{F}_\omega, \Gamma_\omega \rangle$, we obtain alternative proofs of the expressiveness w.r.t. similarity of the logics defined in [1] and [7] (in the unlabelled case), respectively.

We conclude this section with some results concerning the final sequence of a T-relator Γ, in case the hypotheses of Corollary 4.23 are satisfied.

Proposition 4.24 *Let Γ and $\mathcal{F}$ be as in Corollary 4.23*[13]*. Furthermore, assume that there exists a functor $\mathsf{L} : \mathsf{Set} \to \mathsf{Set}$ such that $\mathcal{F}$ takes $\langle X, \mathcal{L}, \models \rangle$ to a language of form $\langle \mathsf{T}X, \mathsf{L}\mathcal{L}, \models' \rangle$, for each $\langle X, \mathcal{L}, \models \rangle \in |\mathsf{Lang}|$. If the final sequence of T stabilises at α, and the initial sequence of L stabilises at $\beta \leq \alpha$, then the final sequence of Γ also stabilises at α.*

[12] Hence, by Proposition 2.7, the relation sequence induced by Γ stabilizes at some α.

[13] In particular, $\mathcal{F}$ preserves expressiveness w.r.t. Γ.

Proof. The construction of colimits in Lang results in the final sequence of $\mathcal{F}$ being of form $(\langle \mathcal{L}_\alpha, Z_\alpha, \models_\alpha \rangle), (\langle \iota^\alpha_\beta, p^\alpha_\beta \rangle)$, where $(\mathcal{L}_\alpha), (\iota^\alpha_\beta)$ is the initial sequence of L, and $(Z_\alpha), (p^\alpha_\beta)$ is the final sequence of T. Moreover, the additional constraints on T and L together with the definition of arrows in Lang ensure that the final sequence of $\mathcal{F}$ also stabilises at α.

On the other hand, by Theorem 4.18, $\geq_{\mathcal{L}_\alpha}$ and $\geq_{\mathcal{L}_{\alpha+1}}$ capture $\gtrsim_\alpha$ and $\gtrsim_{\alpha+1}$, respectively. Hence, for $x, y \in Z_{\alpha+1}$, the following holds:

$$x \gtrsim_{\alpha+1} y \ \text{ iff } \ x \geq_{\mathcal{L}_{\alpha+1}} y \ \text{ iff } \ p^{\alpha+1}_\alpha(x) \geq_{\mathcal{L}_\alpha} p^{\alpha+1}_\alpha(y) \ \text{ iff } \ p^{\alpha+1}_\alpha(x) \gtrsim_\alpha p^{\alpha+1}_\alpha(y)$$

with the second equivalence following from the fact that $\langle \iota^{\alpha+1}_\alpha, p^{\alpha+1}_\alpha \rangle$ defines an isomorphism in Lang. As a result, $p^{\alpha+1}_\alpha : \langle Z_{\alpha+1}, \gtrsim_{\alpha+1} \rangle \to \langle Z_\alpha, \gtrsim_\alpha \rangle$ is an isomorphism in Rel, and hence the final sequence of Γ stabilises at α. □

Thus, Proposition 4.24 allows us to make statements about the degree of accessibility of a T-relator Γ, by exhibiting a language constructor for T which preserves expressiveness w.r.t. Γ. All the examples considered in this paper are such that the functor L of Proposition 4.24 exists.

We now assume that T is ω-accessible. Then, as noted in [14], the final sequence of T stabilises at $\omega + \omega$. Moreover, the maps $p^{\omega+n+1}_{\omega+n}$ with $n = 0, 1, \ldots$ are all injective. Combining this observation with Proposition 4.24 yields the following result.

Corollary 4.25 *Let $\mathsf{T} : \mathsf{Set} \to \mathsf{Set}$ be an ω-accessible endofunctor, let $\Gamma : \mathsf{Rel} \to \mathsf{Rel}$, $\mathcal{F} : \mathsf{Lang} \to \mathsf{Lang}$ and $\mathsf{L} : \mathsf{Set} \to \mathsf{Set}$ be as in Proposition 4.24, and assume that L is ω-accessible. Then:*

- (i) *The final sequence of Γ stabilises at $\omega + \omega$.*
- (ii) *The maps $p^{\omega+n+1}_{\omega+n} : \langle Z_{\omega+n+1}, \gtrsim_{\omega+n+1} \rangle \to \langle Z_{\omega+n+1}, \gtrsim_{\omega+n+1} \rangle$ with $n = 0, 1, \ldots$ are order-reflecting.*

Proof. The fact that L is ω-accessible results in its initial sequence stabilising at ω. The first statement then follows immediately from Proposition 4.24. The second statement follows by an argument similar to the one in the proof of Proposition 4.24. □

Thus, under the hypotheses of Corollary 4.25, the last ω-steps in the final sequence of Γ are determined by the corresponding steps in the final sequence of T. The induced language for coalgebras is not influenced by these steps.

If $\Gamma_\supseteq$ and Γ_ω are as in Example 2.9 and Section 3.1, respectively, it follows from Corollary 4.25 that their final sequences stabilise at $\omega + \omega$.

5 Compositionality

In this section we show that various operations on coalgebraic types induce operations on relators on the one hand, and on language constructors on the other. Moreover, the expressiveness of language constructors w.r.t. given relators is preserved by the induced operations. As a result, notions of similarity for complex coalgebraic types, as well as logics capturing them can be derived in a compositional manner.

We begin by recalling the definition of products and coproducts in Rel. If $R_i \subseteq X_i \times Y_i$ with $i = 1, 2$, then $R_1 \times R_2 \subseteq (X_1 \times X_2) \times (Y_1 \times Y_2)$ and $R_1 + R_2 \subseteq (X_1 + X_2) \times (Y_1 + Y_2)$ are given by:

$$\begin{aligned} (x_1, x_2)\ (R_1 \times R_2)\ (y_1, y_2) &\text{ iff } x_1\, R_1\, y_1 \text{ and } x_2\, R_2\, y_2 \\ \iota_i(x_i)\ (R_1 + R_2)\ \iota_j(y_j) &\text{ iff } i = j \text{ and } x_i\, R_i\, y_i \end{aligned}$$

with $x_i \in X_i$ and $y_i \in Y_i$, for $i = 1, 2$. Similarly to products, one can define, for each relation $R_1 \subseteq X_1 \times Y_1$, a relation $(R_1)^A \subseteq (X_1)^A \times (Y_1)^A$ by:

$$(x_a)_{a \in A}\, (R_1)^A\, (y_a)_{a \in A} \text{ iff } x_a\, R_1\, y_a \text{ for all } a \in A$$

with $x_a \in X_1$ for $a \in A$. The above operations on relations can be used to derive $(\mathsf{T}_1 \times \mathsf{T}_2)$-, $(\mathsf{T}_1 + \mathsf{T}_2)$- and $(\mathsf{T}_1)^A$-relators from T_1- and T_2-relators.

Definition 5.1 Let Γ_1 and Γ_2 be relators for T_1 and T_2, respectively. Define $\Gamma_1 \oplus \Gamma_2$, $\Gamma_1 \otimes \Gamma_2$, $(\Gamma_1)^A : \mathsf{Rel} \to \mathsf{Rel}$ by:

- $R \subseteq X \times Y \xmapsto{\Gamma_1 \oplus \Gamma_2} \Gamma_1(R) + \Gamma_2(R) \subseteq (\mathsf{T}_1 + \mathsf{T}_2)X \times (\mathsf{T}_1 + \mathsf{T}_2)Y$
- $R \subseteq X \times Y \xmapsto{\Gamma_1 \otimes \Gamma_2} \Gamma_1(R) \times \Gamma_2(R) \subseteq (\mathsf{T}_1 \times \mathsf{T}_2)X \times (\mathsf{T}_1 \times \mathsf{T}_2)Y$
- $R \subseteq X \times Y \xmapsto{(\Gamma_1)^A} \Gamma_1(R)^A \subseteq (\mathsf{T}_1 X)^A \times (\mathsf{T}_1 Y)^A$.

In addition, relators can be combined using functor composition.

Proposition 5.2 $\Gamma_1 \circ \Gamma_2$, $\Gamma_1 \oplus \Gamma_2$, $\Gamma_1 \otimes \Gamma_2$, $(\Gamma_1)^A$ *are relators for* $\mathsf{T}_1 \circ \mathsf{T}_2$, $\mathsf{T}_1 + \mathsf{T}_2$, $\mathsf{T}_1 \times \mathsf{T}_2$ *and* $(\mathsf{T}_1)^A$, *respectively.*

This allows us to derive relators (and therefore notions of simulation) for combinations of coalgebraic types from relators for the types being combined.

Example 5.3 Let $\Gamma_\supseteq$ be as in Example 2.9. A relation $R \subseteq C \times D$ is a $(\Gamma_\supseteq)^A$-simulation between $(\mathcal{P}_\omega)^A$-coalgebras $\langle C, \gamma \rangle$ and $\langle D, \delta \rangle$ if, whenever $c\, R\, d$ and $d' \in \delta(d)(a)$ for some $a \in A$, there exists $c' \in \gamma(c)(a)$ such that $c'\, R\, d'$. Moreover, it is shown in [10] that $(\Gamma_\supseteq)^A$-similarity coincides with trace inclusion:

$c \gtrsim d$ if and only if $\mathsf{traces}(c) \supseteq \mathsf{traces}(d)$, where, for $c \in C$, $\mathsf{traces}(c)$ consists of all finite sequences $(a_1, \ldots, a_n)$ of elements of A, such that there exist $c_0, c_1, \ldots, c_n \in C$ with $c_0 = c$ and $c_i \in \gamma(c_{i-1})(a_i)$ for $i = 1, \ldots, n$.

Example 5.4 Let Γ_ω be as in Section 3.1. A relation $R \subseteq C \times C$ is a $(\Gamma_\omega)^A$-simulation on an $(\mathcal{S}_\omega)^A$-coalgebra $\langle C, \gamma \rangle$ if, whenever $c\,R\,d$, $a \in A$ and $X \subseteq C$ is R^{op}-closed, we have $\gamma(c)(a)[X] \geq \gamma(d)(a)[X]$. Thus, $(\Gamma_\omega)^A$-simulation coincides with the notion of simulation defined in [7] (see Remark 3.10).

Next, we show how language constructors for combinations of coalgebraic types can be obtained by combining language constructors for the component types.

Definition 5.5 Let $\mathcal{F}_1$ and $\mathcal{F}_2$ be language constructors for T_1 and T_2, respectively. Define $\mathcal{F}_1 \oplus \mathcal{F}_2$, $\mathcal{F}_1 \otimes \mathcal{F}_2$, $(\mathcal{F}_1)^A : \mathsf{Lang} \to \mathsf{Lang}$ by [14] [15]:

- $\langle X, \mathcal{L}, \models \rangle \xmapsto{\mathcal{F}_1 \oplus \mathcal{F}_2} \langle (\mathsf{T}_1 + \mathsf{T}_2)X, \{ \langle \kappa_i \rangle \varphi \mid \varphi \in \mathcal{L}_i \}, (\mathsf{Gr}(\iota_1) \circ \models_1 \circ \mathsf{Gr}(\iota_1^{-1})) \cup (\mathsf{Gr}(\iota_2) \circ \models_2 \circ \mathsf{Gr}(\iota_2^{-1})) \rangle$
- $\langle X, \mathcal{L}, \models \rangle \xmapsto{\mathcal{F}_1 \otimes \mathcal{F}_2} \langle (\mathsf{T}_1 \times \mathsf{T}_2)X, \{ [\pi_i]\varphi \mid \varphi \in \mathcal{L}_i \}, (\mathsf{Gr}(\iota_1) \circ \models_1 \circ \mathsf{Gr}(\pi_1)) \cup (\mathsf{Gr}(\iota_2) \circ \models_2 \circ \mathsf{Gr}(\pi_2)) \rangle$
- $\langle X, \mathcal{L}, \models \rangle \xmapsto{(\mathcal{F}_1)^A} \langle (\mathsf{T}_1 X)^A, \{ [a]\varphi \mid \varphi \in \mathcal{L}_1 \}, \bigcup_{a \in A} (\mathsf{Gr}(\iota_a) \circ \models_1 \circ \mathsf{Gr}(\pi_a)) \rangle$

if $\langle X, \mathcal{L}, \models \rangle \xmapsto{\mathcal{F}_i} \langle \mathsf{T}_i X, \mathcal{L}_i, \models_i \rangle$ for $i = 1, 2$.

We note that the modal operators $[\pi_i]$ and $\langle \kappa_i \rangle$ with $i = 1, 2$ and $[a]$ with $a \in A$ are similar to the ones used in [9].

Remark 5.6 Since the set of formulae of $(\mathcal{F}_1 \oplus \mathcal{F}_2)\langle X, \mathcal{L}, \models \rangle$ is (isomorphic to) the coproduct $\mathcal{L}_1 + \mathcal{L}_2$, the associated satisfaction relation could equivalently be defined as the coproduct $\models_1 + \models_2$ in Rel. We have chosen the more complex formulation in Definition 5.5 for coherence with the definitions of $\mathcal{F}_1 \otimes \mathcal{F}_2$ and $(\mathcal{F}_1)^A$. An alternative approach to defining $\mathcal{F}_1 \otimes \mathcal{F}_2$ and $(\mathcal{F}_1)^A$ would be to take $\mathcal{L}_1 \times \mathcal{L}_2$ and $(\mathcal{L}_1)^A$ as sets of formulae in $(\mathcal{F}_1 \otimes \mathcal{F}_2)\langle X, \mathcal{L}, \models \rangle$ and $(\mathcal{F}_1)^A \langle X, \mathcal{L}, \models \rangle$, respectively, in which case the corresponding satisfaction relations would be given by $\models_1 \times \models_2$ and $(\models_1)^A$, respectively. The reason for not taking this approach is that, for A infinite, this yields infinitary modal operators.

Language constructors can also be combined using functor composition.

[14] Similar operations on language constructors were defined in [5].

[15] In each case, the formula $\top$, with the required interpretation, is also added to the resulting language.

Proposition 5.7 *$\mathcal{F}_1 \circ \mathcal{F}_2$, $\mathcal{F}_1 \oplus \mathcal{F}_2$, $\mathcal{F}_1 \otimes \mathcal{F}_2$, $(\mathcal{F}_1)^A$ are language constructors for $\mathsf{T}_1 \circ \mathsf{T}_2$, $\mathsf{T}_1 + \mathsf{T}_2$, $\mathsf{T}_1 \times \mathsf{T}_2$ and $(\mathsf{T}_1)^A$, respectively.*

Example 5.8 Let $\mathcal{F}_\supseteq$ be as in Example 4.6. Then, $(\mathcal{F}_\supseteq)^A$ takes a language $\langle X, \mathcal{L}, \models \rangle$ to the language $\langle (\mathcal{P}_\omega X)^A, \mathcal{L}', \models' \rangle$, where $\mathcal{L}'$ is generated in two steps by the following syntax:

$$\begin{array}{rcll} \mathcal{L}' \ni \varphi' & ::= & \top \mid [a]\rho & (\rho \in \mathcal{L}_0) \\ \mathcal{L}_0 \ni \rho & ::= & \top \mid \Diamond\varphi \mid \rho_1 \wedge \rho_2 & (\varphi \in \mathcal{L}) \end{array}$$

and where $\models' \subseteq (\mathcal{P}_\omega X)^A \times \mathcal{L}'$ is defined in two steps by:

$$\begin{array}{lll} f \models' [a]\rho \ \text{ iff } \ f(a) \models_0 \rho & & (f \in \mathcal{P}_\omega(X)^A) \\ Y \models_0 \Diamond\phi \ \text{ iff } \ \exists x \in Y.\, x \models \varphi & & (Y \in \mathcal{P}_\omega(X)) \end{array}$$

Since the modal operator $[a]$ distributes over conjunctions, the language induced by $\langle (\mathcal{F}_\supseteq)^A, (\Gamma_\supseteq)^A \rangle$ is equivalent to a fragment of Hennessy-Milner logic (with $[a]\top$ being semantically equivalent to $\top$, and with $[a]\Diamond\varphi$ being semantically equivalent to $\langle a \rangle \varphi'$ whenever φ is semantically equivalent to φ').

Example 5.9 Let $\mathcal{F}_\omega$ be as in Section 4.3. It then follows by an argument similar to the one in Example 5.8 that the language induced by $\langle (\mathcal{F}_\omega)^A, (\Gamma_\omega)^A \rangle$ is equivalent to a fragment of the language used in [7] (with a formula of form $[a]\Diamond_p\varphi$ corresponding to a formula of form $\langle a \rangle_p \varphi'$).

Our next result shows that the expressivity condition required to derive expressive logics for simulation is preserved by the previously-defined operations on relators and language constructors, respectively.

Proposition 5.10 *If $\mathcal{F}_i$ preserves expressiveness w.r.t. Γ_i, for $i = 1, 2$, then $\mathcal{F}_1 \circ \mathcal{F}_2$, $\mathcal{F}_1 \oplus \mathcal{F}_2$, $\mathcal{F}_1 \otimes \mathcal{F}_2$ and $(\mathcal{F}_1)^A$ preserve expressiveness w.r.t. $\Gamma_1 \circ \Gamma_2$, $\Gamma_1 \oplus \Gamma_2$, $\Gamma_1 \otimes \Gamma_2$ and $(\Gamma_1)^A$, respectively.*

Proof (Sketch). In the case of $\mathcal{F}_1 \circ \mathcal{F}_2$, the statement follows immediately from the definition of preservation of expressiveness w.r.t. a relator. Of the remaining cases, we only consider that of coproducts. (The other two are treated similarly.) Let $\langle X, \mathcal{L}, \models \rangle$ be expressive for $\langle X, R \rangle$. Hence, $\mathcal{F}_i \langle X, \mathcal{L}, \models \rangle = \langle \mathsf{T}_i X, \mathcal{L}_i, \models_i \rangle$ is expressive for $\langle \mathsf{T}_i X, \Gamma_i R \rangle$. Now let $i, j \in \{1, 2\}$, $t_i \in \mathsf{T}_i X$ and $s_j \in T_j X$. If $i \neq j$, the fact that $\iota_i(t_i)$ and $\iota_j(s_j)$ are not related by $\Gamma_1 R + \Gamma_2 R$ is witnessed by the formula $\langle \kappa_j \rangle \top$, which holds in $\iota_j(s_j)$ but not in $\iota_i(t_i)$. If $i = j$, the fact that $\iota_i(t_i)$ and $\iota_i(s_i)$ are not related by $\Gamma_1 R \times \Gamma_2 R$ (and therefore t_i and s_i are not related by $\Gamma_i R$) is witnessed by the formula $\langle \kappa_i \rangle \varphi_i$, where φ_i holds in s_i but not in t_i. □

Example 5.11 Taking $\langle \mathcal{F}, \Gamma \rangle = \langle (\mathcal{F}_{\supseteq})^A, (\Gamma_{\supseteq})^A \rangle$ in Corollary 4.23 yields a language which characterizes trace inclusion. Its syntax is given by:

$$\begin{aligned} \mathcal{L} \ni \varphi \ &::= \ \top \mid [a]\rho && (\rho \in \mathcal{L}_0) \\ \mathcal{L}_0 \ni \rho \ &::= \ \top \mid \Diamond\varphi \mid \rho_1 \wedge \rho_2 && (\varphi \in \mathcal{L}) \end{aligned}$$

while its coalgebraic semantics is defined inductively by:

$$\begin{aligned} f \models [a]\rho \ &\text{iff} \ f(a) \models_0 \rho && (f \in \mathcal{P}_\omega(C)^A) \\ Y \models_0 \Diamond\varphi \ &\text{iff} \ \exists\, c \in Y.\, c \models_\gamma \varphi && (Y \in \mathcal{P}_\omega(C)) \\ c \models_\gamma \varphi \ &\text{iff} \ \gamma(c) \models \varphi && (c \in C) \end{aligned}$$

with $\langle C, \gamma \rangle$ a $(\mathcal{P}_\omega)^A$-coalgebra.

Example 5.12 Similarly, taking $\langle \mathcal{F}, \Gamma \rangle = \langle (\mathcal{F}_\omega)^A, (\Gamma_\omega)^A \rangle$ yields a language which characterizes probabilistic simulation. Its syntax is given by:

$$\begin{aligned} \mathcal{L} \ni \varphi \ &::= \ \top \mid [a]\rho && (\rho \in \mathcal{L}_0) \\ \mathcal{L}_0 \ni \rho \ &::= \ \top \mid \Diamond_p\varphi \mid \rho_1 \wedge \rho_2 \mid \rho_1 \vee \rho_2 && (\varphi \in \mathcal{L}) \end{aligned}$$

while its coalgebraic semantics is defined inductively by:

$$\begin{aligned} f \models [a]\rho \ &\text{iff} \ f(a) \models_0 \rho && (f \in \mathcal{S}_\omega(C)^A) \\ \mu \models_0 \Diamond_p\varphi \ &\text{iff} \ \mu[[\![\varphi]\!]_\gamma] \geq p && (\mu \in \mathcal{S}_\omega(C)) \\ c \models_\gamma \varphi \ &\text{iff} \ \gamma(c) \models \varphi && (c \in C) \end{aligned}$$

with $\langle C, \gamma \rangle$ an $(\mathcal{S}_\omega)^A$-coalgebra.

Finally, it is also possible to define language constructors for constant and identity functors: in the case of the constant functor $X \mapsto A$, the language constructor provides an atomic formula a for each $a \in A$, whereas in the case of the identity functor, the language constructor is itself the identity (on Lang). Each of these language constructors preserves expressiveness w.r.t. the corresponding minimal relator. As a result, the compositional techniques described in this section can be applied to any functor T of form:

$$\mathsf{T} ::= A \mid \mathsf{Id} \mid \mathcal{P}_\omega \mid \mathcal{S}_\omega \mid \mathsf{T}_1 \circ \mathsf{T}_2 \mid \mathsf{T}_1 + \mathsf{T}_2 \mid \mathsf{T}_1 \times \mathsf{T}_2 \mid (\mathsf{T}_1)^A$$

in order to derive both a notion of simulation for T-coalgebras, and a logic which captures this notion of simulation. This yields notions of simulation and corresponding logics for a variety of probabilistic system types (see e.g. [2] for a survey of probabilistic system types studied in the literature).

6 Conclusions

We have presented an inductive method for defining logics which capture simulation. We used relators to define notions of simulation for coalgebraic types, language constructors to formalise one step in the definition of languages for coalgebras, and an expressivity condition involving a language constructor and a relator to ensure the expressiveness of the induced languages w.r.t. the induced notions of simulation. This method was applied to obtain Baltag's logics for coalgebraic simulation, as well as logics capturing simulation on unlabelled (probabilistic) transition systems.

We have also shown that various operations on coalgebraic signatures induce corresponding operations on relators as well as on language constructors, with the expressivity condition being preserved by the induced operations. This has resulted in compositional techniques for defining notions of simulation and logics which capture simulation. Such techniques were used to obtain a coalgebraic characterization of simulation on discrete Markov processes, as well as a logic which captures this notion of simulation.

Our approach can also be used to derive notions of simulation and suitable logics for other probabilistic system types, including types which combine nondeterminism and probability (through a combination of $\mathcal{P}_\omega$ and $\mathcal{S}_\omega$ in the signature functor). The study of the resulting logics and of their relevance to system specification is the subject of ongoing work.

References

[1] A. Baltag. A logic for coalgebraic simulation. In H. Reichel, editor, *Coalgebraic Methods in Computer Science*, volume 33 of *Electronic Notes in Theoretical Computer Science*. Elsevier, 2000.

[2] F. Bartels, A. Sokolova, and E. de Vink. A hierarchy of probabilistic system types. In H.P. Gumm, editor, *Coalgebraic Methods in Computer Science*, volume 82.1 of *Electronic Notes in Theoretical Computer Science*, 2003.

[3] F. Borceux. *Handbook of Categorical Algebra*, volume II. Cambridge University Press, Cambridge, 1994.

[4] A. Carboni, G.M. Kelly, and R.J. Wood. A 2-categorical approach to change of base and geometric morphisms I. Technical Report 90-1, Department of Pure Mathematics, University of Sidney, 1990.

[5] C. Cîrstea. On expressivity and compositionality in logics for coalgebras. In H.P. Gumm, editor, *Coalgebraic Methods in Computer Science*, volume 82.1 of *Electronic Notes in Theoretical Computer Science*. Elsevier, 2003.

[6] E.P. de Vink and J.J.M.M. Rutten. Bisimulation for probabilistic transition systems: a coalgebraic approach. *Theoretical Computer Science*, 221:271–293, 1999.

[7] J. Desharnais. A logical characterisation of simulation for labelled Markov chains. In *Proceedings of PROBMIV'99*. University of Birmingham, 1999. Technical Report CS-99-8.

[8] W.H. Hesselink and A. Thijs. Fixpoint semantics and simulation. *Theoretical Computer Science*, 238:275–311, 2000.

[9] B. Jacobs. Many-sorted coalgebraic modal logic: a model-theoretic study. *Theoretical Informatics and Applications*, 35, 2001.

[10] B. Jacobs and J. Hughes. Simulations in coalgebra. In H.P. Gumm, editor, *Coalgebraic Methods in Computer Science*, volume 82.1 of *Electronic Notes in Theoretical Computer Science*. Elsevier, 2003.

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

[12] L.S. Moss. Coalgebraic logic. *Annals of Pure and Applied Logic*, 96:277–317, 1999.

[13] A. Thijs. *Simulation and Fixpoint Semantics*. PhD thesis, Rijksuniversiteit Groningen, 1996.

[14] J. Worrell. Terminal sequences for accessible endofunctors. In B. Jacobs and J. Rutten, editors, *Coalgebraic Methods in Computer Science*, volume 19 of *Electronic Notes in Theoretical Computer Science*. Elsevier, 1999.