Coalgebraic Symbolic Semantics*

Filippo Bonchi[1,2] and Ugo Montanari[1]

[1] Dipartimento di Informatica, Università di Pisa
[2] Centrum voor Wiskunde en Informatica (CWI)

Abstract. The operational semantics of interactive systems is usually described by labeled transition systems. Abstract semantics (that is defined in terms of bisimilarity) is characterized by the final morphism in some category of coalgebras. Since the behaviour of interactive systems is for many reasons infinite, *symbolic semantics* were introduced as a mean to define smaller, possibly finite, transition systems, by employing symbolic actions and avoiding some sources of infiniteness. Unfortunately, symbolic bisimilarity has a different "shape" with respect to ordinary bisimilarity, and thus the standard coalgebraic characterization does not work. In this paper, we introduce its coalgebraic models.

1 Introduction

A compositional interactive system is usually defined as a labelled transition system (LTS) where states are equipped with an algebraic structure. Abstract semantics is often defined as bisimilarity. Then a key property is that bisimilarity be a congruence, i.e. that abstract semantics respects the algebraic operations.

Universal Coalgebra [1] provides a categorical framework where the behaviour of dynamical systems can be characterized as *final semantics*. More precisely, if $\mathbf{Coalg_B}$ (i.e., the category of $\mathbf{B}$-coalgebras and $\mathbf{B}$-cohomomorphisms for a certain endofunctor $\mathbf{B}$) has a final object, then the behavior of a $\mathbf{B}$-coalgebra is defined as a final morphism. Intuitively, a final object is a universe of abstract behaviors and a final morphism is a function mapping each system in its abstract behavior. Ordinary LTSs can be represented as coalgebras for a suitable functor. Then, two states are bisimilar if and only if they are identified by a final morphism. The image of a certain LTS through a final morphism is its minimal representative (with respect to bisimilarity), which in the finite case can be computed via the list partitioning algorithm [2].

When bisimilarity is not a congruence, the abstract semantics is defined either as the *largest congruence contained into bisimilarity* [3] or as the *largest bisimulation that is also a congruence* [4]. In this paper we focus on the latter and we call it *saturated bisimilarity* ($\sim^S$). Indeed it coincides with ordinary bisimilarity on the *saturated transition system*, that is obtained by the original LTS by adding the transition $p \xrightarrow{c,a} q$, for every context c, whenever $c(p) \xrightarrow{a} q$.

* This work was carried out during the tenure of an ERCIM "Alain Bensoussan" Fellowship Programme and supported by the IST 2004-16004 SEnSOria.

A. Kurz, M. Lenisa, and A. Tarlecki (Eds.): CALCO 2009, LNCS 5728, pp. 173–190, 2009.

Many interesting abstract semantics are defined in this way. For example, since late and early bisimilarity of π-calculus [5] are not preserved under substitution (and thus under input prefixes), in [6] Sangiorgi introduces *open bisimilarity* as the largest bisimulation on π-calculus agents which is closed under substitutions. Other noteworthy examples are asynchronous π-calculus [7,8] and mobile ambients calculus [9,10]. The definition of saturated bisimilarity as ordinary bisimulation on the saturated LTS, while in principle operational, often makes the portion of LTS reachable by any nontrivial agent infinite state, and in any case is very inefficient, since it introduces a large number of additional states and transitions. Inspired by [11], Sangiorgi defines in [6] a symbolic transition system and symbolic bisimilarity that efficiently characterizes open bisimilarity. After this, many formalisms have been equipped with a symbolic semantics.

In [12], we have introduced a general model that describes at an abstract level both saturated and symbolic semantics. In this abstract setting, a symbolic transition $p \xrightarrow{c,\alpha}_{\beta} p'$ means that $c(p) \xrightarrow{\alpha} p'$ and c is a smallest context that allows p to performs such transition. Moreover, a certain *derivation relation* $\vdash$ amongst the transitions of a systems is defined: $p \xrightarrow{c_1,\alpha_1} p_1 \vdash p \xrightarrow{c_2,\alpha_2} p_2$ means that the latter transition is a logical consequence of the former. In this way, if all and only the saturated transitions are logical consequences of symbolic transitions, then saturated bisimilarity can be retrieved via the symbolic LTS.

However, the ordinary bisimilarity over the symbolic transition system differs from saturated bisimilarity. Symbolic bisimilarity is thus defined with an asymmetric shape. In the bisimulation game, when a player proposes a transition, the opponent can answer with a move with a different label. For example in the open π-calculus, a transition $p \xrightarrow{[a=b],\tau} p'$ can be matched by $q \xrightarrow{\tau} q'$. Moreover, the bisimulation game does not restart from p' and q', but from p' and $q'\{b/a\}$.

For this reason, ordinary coalgebras fail to characterize symbolic bisimilarity. Here, we provide coalgebraic models for it by relying on the framework of [12].

Consider the example of open bisimilarity discussed above. The fact that open bisimulation does not relate the arriving states p' and q', but p' and $q'\{b/a\}$, forces us to look for models equipped with an algebraic structure. In [13], *bialgebras* are introduced as a both algebraic and coalgebraic model, while an alternative approach based on *structured coalgebras*, i.e. on coalgebras in categories of algebras, is presented in [14]. In this paper we adopt the latter and we introduce $\mathbf{Coalg_H}$ (Sec. 4), a category of structured coalgebras where the saturated transition system can be naively modeled in such a way that $\sim^S$ coincides with the kernel of a final morphism. Then, we focus only on those $\mathbf{H}$-coalgebras whose sets of transitions are closed w.r.t. the derivation relation $\vdash$. These form the category of *saturated coalgebras* $\mathbf{Coalg_{S_T}}$ (Sec. 5.1) that is a covariety of $\mathbf{Coalg_H}$. Thus, it has a final object and bisimilarity coincides with the one in $\mathbf{Coalg_H}$.

In order to characterize symbolic bisimilarity, we introduce the notions of *redundant transition* and *semantically redundant transition*. Intuitively, a transition $p \xrightarrow{c_2,\alpha_2} q$ is redundant if there exists another transition $p \xrightarrow{c_1,\alpha_1} p_1$ that logically implies it, that is $p \xrightarrow{c_1,\alpha_1} p_1 \vdash p \xrightarrow{c_2,\alpha_2} q$, while it is semantically redundant, if it is redundant up to bisimilarity, i.e., $p \xrightarrow{c_1,\alpha_1} p_1 \vdash p \xrightarrow{c_2,\alpha_2} p_2$ and

q is bisimilar to p_2. Now, in order to retrieve saturated bisimilarity by disregarding redundant transitions, we have to remove from the saturated transition system not only all the redundant transitions, but also the semantically redundant ones. This is done in the category of *normalized coalgebras* $\mathbf{Coalg}_{\mathbf{N_T}}$ (Sec. 5.2). These are defined as coalgebras without redundant transitions. Thus, by definition, a final coalgebra in $\mathbf{Coalg}_{\mathbf{N_T}}$ has no semantically redundant transitions. The main peculiarity of $\mathbf{Coalg}_{\mathbf{N_T}}$ relies in its morphisms. Indeed, ordinary (cohomo)morphisms between LTSs must preserve and reflect all the transitions ("zig-zag" morphisms), while morphisms between normalized coalgebras must preserve only those transitions that are not semantically redundant.

Moreover, we prove that $\mathbf{Coalg}_{\mathbf{S_T}}$ and $\mathbf{Coalg}_{\mathbf{N_T}}$ are isomorphic (Sec. 5.3). This means that a final morphism in the latter category still characterizes $\sim^S$, but with two important differences w.r.t. $\mathbf{Coalg}_{\mathbf{S_T}}$. First of all, in a final $\mathbf{N_T}$-coalgebra, there are not semantically redundant transitions. Intuitively, a final $\mathbf{N_T}$-coalgebra is a universe of *abstract symbolic behaviours* and a final morphism maps each system in its abstract symbolic behaviour. Secondly, minimization in $\mathbf{Coalg}_{\mathbf{N_T}}$ is feasible, while in $\mathbf{Coalg}_{\mathbf{S_T}}$ is not, because saturated coalgebras have all the redundant transitions. Minimizing in $\mathbf{Coalg}_{\mathbf{N_T}}$ coincides with a *symbolic minimization algorithm* that we have introduced in [15] (Sec. 6). The algorithm shows another peculiarity of normalized coalgebras: minimization relies on the algebraic structure. Since in bialgebras bisimilarity abstracts away from this, we can conclude that our normalized coalgebras are not bialgebras. This is the reason why we work with structured coalgebras.

The background is in Sec. 2 and 3.

2 Saturated and Symbolic Semantics

In this section we recall the general framework for symbolic bisimilarity that we have introduced in [12]. As running example, we will use open Petri nets [16]. However, our theory has as special cases the abstract semantics of many formalisms such as ambients [9], open [6] and asynchronous [7] π-calculus.

2.1 Saturated Semantics

Given a small category $\mathbf{C}$, a $\Gamma(\mathbf{C})$-algebra is an algebra for the algebraic specification in Fig. 1(A) where $|\mathbf{C}|$ denotes the set of objects of $\mathbf{C}$, $||\mathbf{C}||$ the set of arrows of $\mathbf{C}$ and, for all $i, j \in |\mathbf{C}|$, $\mathbf{C}[i, j]$ denotes the set of arrows from i to j.

Thus, a $\Gamma(\mathbf{C})$-algebra $\mathbb{X}$ consists of a $|\mathbf{C}|$-sorted family $X = \{X_i \mid i \in |\mathbf{C}|\}$ of sets and a function $c_{\mathbb{X}} : X_i \to X_j$ for all $c \in \mathbf{C}[i, j]$.[1]

The main definition of the framework presented in [12] is that of *context interactive systems*. In our theory, an interactive system is a state-machine that can interact with the environment (contexts) through an evolving interface.

[1] Note that $\Gamma(\mathbf{C})$-algebras coincide with functors from $\mathbf{C}$ to $\mathbf{Set}$ and $\Gamma(\mathbf{C})$-homomorphisms coincide with natural transformations amongst functors. Thus, $\mathbf{Alg}_{\Gamma(\mathbf{C})}$ is isomorphic to $\mathbf{Set}^{\mathbf{C}}$(the category of covariant presheaves over $\mathbf{C}$).

specification $\Gamma(\mathbf{C}) =$
sorts
$i \qquad \forall i \in |\mathbf{C}|$
operations
$c : i \rightarrow j \qquad \forall c \in \mathbf{C}[i,j]$
equations
$id_i(x) = x$
$e(d(x)) = c(x) \qquad \forall d; e = c$

(A)

$$(\text{IN})\,\frac{i \in I}{N, m \xrightarrow{+i} N, m \oplus i}$$

$$(\text{TR})\,\frac{t \in T \quad l(t) = \lambda \quad m =^{\bullet} t \oplus c}{N, m \xrightarrow{\lambda} N, t^{\bullet} \oplus c}$$

(B)

Fig. 1. (A) Algebraic specification $\Gamma(\mathbf{C})$. (B) Operational semantics of open nets.

Definition 1 (Context Interactive System). *A* context interactive system *$\mathcal{I}$ is a quadruple $\langle \mathbf{C}, \mathbb{X}, O, tr \rangle$ where:*

- *$\mathbf{C}$ is a category,*
- *$\mathbb{X}$ is a $\Gamma(\mathbf{C})$-algebra,*
- *O is a set of observations,*
- *$tr \subseteq X \times O \times X$ is a labeled transition relation ($p \xrightarrow{o} p'$ means $(p, o, p') \in tr$).*

Intuitively, objects of $\mathbf{C}$ are interfaces of the system, while arrows are contexts. Every element p of X_i represents a state with interface i and it can be inserted into the context $c \in \mathbf{C}[i,j]$, obtaining a new state $c_{\mathbb{X}}(p)$ that has interface j. Every state can evolve into a new state (possibly with different interface) producing an observation $o \in O$.

The abstract semantics of interactive systems is usually defined through behavioural equivalences. In [12] we proposed a general notion of bisimilarity that generalizes the abstract semantics of a large variety of formalisms [9,7,6,17,18,19]. The idea is that two states of a system are equivalent if they are indistinguishable from an external observer that, in any moment of their execution, can insert them into some environment and then observe some transitions.

Definition 2 (Saturated Bisimilarity). *Let $\mathcal{I} = \langle \mathbf{C}, \mathbb{X}, O, tr \rangle$ be a context interactive system. Let $R = \{R_i \subseteq X_i \times X_i \mid i \in |\mathbf{C}|\}$ be a $|\mathbf{C}|$-sorted family of symmetric relations. R is a* saturated bisimulation *iff, $\forall i, j \in |\mathbf{C}|$, $\forall c \in \mathbf{C}[i,j]$, whenever pR_iq:*

- *$c_{\mathbb{X}}(p)\, R_j\, c_{\mathbb{X}}(q)$,*
- *if $p \xrightarrow{o} p'$, then $q \xrightarrow{o} q'$ and $p' R_k q'$ (for some $k \in |\mathbf{C}|$).*

We write $p \sim^S_i q$ iff there is a saturated bisimulation R such that pR_iq.

An alternative but equivalent definition can be given by defining the *saturated transition system* (SATTS) as follows: $p \xrightarrow{c,o}_S q$ if and only if $c(p) \xrightarrow{o} q$. Trivially the ordinary bisimilarity over SATTS coincides with $\sim^S$.

Proposition 1. *$\sim^S$ is the coarsest bisimulation congruence.*

2.2 Running Example: Open Petri Nets

Differently from process calculi, Petri nets have not a widely known interactive behaviour. Indeed they model concurrent systems that are closed, in the sense that they do not interact with the environment. *Open nets* [16] are P/T Petri nets that can interact by exchanging tokens on *input* and *output places.*

Definition 3 (Open Net). *An* open net *is a tuple* $N = (S, T, pre, post, l, I, O)$ *where* S *and* T *are the sets of places and transitions* ($S \cap T = \varnothing$)*;* $pre, post : T \to S^{\oplus}$ *are functions mapping each transition to its pre- and post-set;* $l : T \to \Lambda$ *is a labeling function (*Λ *is a set of labels) and* $I, O \subseteq S$ *are the sets of input and output places. A* marked open net *is a pair* $\langle N, m \rangle$ *where* N *is an open net and* $m \in S^{\oplus}$ *is a marking.*

Fig.2 shows five open nets where, as usual, circles represents places and rectangles transitions (labeled with α, β). Arrows from places to transitions represent *pre*, while arrows from transitions to places represent *post*. Input places are denoted by ingoing edges, thus the only input place of N_1 is \$. To make examples easier, hereafter we only consider *open input nets*, i.e., open nets without output places.The operational semantics of marked open nets is expressed by the rules on Fig.1(B), where we use ${}^{\bullet}t$ and $t^{\bullet}$ to denote $pre(t)$ and $post(t)$. The rule (TR) is the standard rule of P/T nets (seen as multisets rewriting). The rule (IN) states that in any moment a token can be inserted inside an input place and, for this reason, the LTS has always an infinite number of states. Fig.2(A) shows part of the infinite transition system of $\langle N_1, a \rangle$. The abstract semantics (denoted by $\sim^N$) is defined in [20] as the ordinary bisimilarity over such an LTS. It is worth noting that $\sim^N$ can be seen as an instance of saturated semantics, where multisets over open places are contexts and transitions are only those generated by the rule (TR).

In the following we formally define $\mathcal{N} = \langle \mathbf{Tok}, \mathbb{N}, \Lambda, tr_{\mathcal{N}} \rangle$ that is the context interactive system of all open nets (labeled over the set of labels Λ).

The category **Tok** is formally defined as follows,

- $|\mathbf{Tok}| = \{I \mid I \text{ is a set of places}\}$,
- $\forall I \in |\mathbf{Tok}|$, $\mathbf{Tok}[I, I] = I^{\oplus}$, $id_I = \varnothing$ and $\forall i_1, i_2 \in I^{\oplus}$, $i_1; i_2 = i_1 \oplus i_2$.

Intuitively objects are sets of input places I, while arrows are multisets of tokens on the input places. We say that a marked open net $\langle N, m \rangle$ has interface I if the set of input places of N is I. For example the marked open net $\langle N_1, a \rangle$ has interface $\{\$\}$. Let us define the $\Gamma(\mathbf{Tok})$-algebra $\mathbb{N}$. For any sort I, the carrier set N_I contains all the marked open nets with interface I. For any operator $i \in \mathbf{Tok}[I, I]$, the function $i_{\mathbb{N}}$ maps $\langle N, m \rangle$ into $\langle N, m \oplus i \rangle$.

The transition structure $tr_{\mathcal{N}}$ (denoted by $\to_{\mathcal{N}}$) associates to a state $\langle N, m \rangle$ the transitions obtained by using the rule (TR) of Fig.1(B). In [12], it is proved that saturated bisimilarity for $\mathcal{N}$ coincides with $\sim^N$. In the remainder of the paper we will use as running example the open nets in Fig.2. Since all the places have different names (with the exception of \$), in order to make lighter the

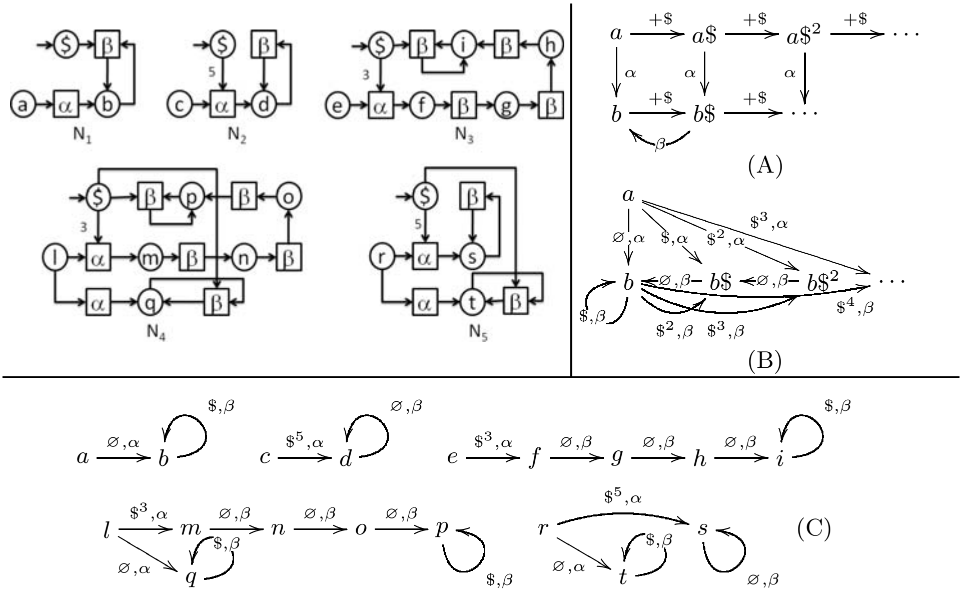


Fig. 2. The open nets N_1, N_2, N_3, N_4 and N_5.(A)Part of the infinite transition system of $\langle N_1, a\rangle$. (B)Part of the infinite saturated transition system of $\langle N_1, a\rangle$.(C)The symbolic transition systems of $\langle N_1, a\rangle$,$\langle N_2, c\rangle$,$\langle N_3, e\rangle$,$\langle N_4, l\rangle$ and $\langle N_5, r\rangle$.

notation, we write only the marking to mean the corresponding marked net, e.g. $b^2\$$ means the marked net $\langle N_1, b^2\$\rangle$.

The marked net a (i.e., $\langle N_1, a\rangle$) represents a system that provides a service β. After the activation α, it provides β whenever the client pay one \$ (i.e., the environment insert a token into \$). The marked net c instead requires five \$ during the activation, but then provides the service β for free. The marked net e, requires three \$ during the activation. For three times, the service β is performed for free and then it costs one \$. It is easy to see that all these marked nets are not bisimilar. Indeed, a client that has only one \$ can have the service β only with a, while a client with five \$ can have the service β for six times only with c. The marked net r represents a system that offers the behaviour of both a and c, i.e. either the activation α is for free and then the service β costs one, or the activation costs five and then the service is for free. Also this marked net is different from all the others.

Now consider the marked net l. It offers the behaviour of both a and e, but it is equivalent to a, i.e. $l \sim^N a$. Roughly, the behaviour of e is absorbed by the behaviour of a. This is analogous to what happens in asynchronous π-calculus [7] where it holds that $a(x).(\overline{a}x \mid p) + \tau.p \sim \tau.p$.

2.3 Symbolic Semantics

Saturated bisimulation is a good notion of equivalence but it is hard to check, since it involves a quantification over all contexts. In [12], we have introduced a general notion of *symbolic bisimilarity* that coincides with saturated bisimilarity, but it avoids to consider all contexts. The idea is to define a symbolic transition system where transitions are labeled both with the usual observation and also with the minimal context that allows the transition.

Definition 4 (Symbolic Context Transition System). *A* symbolic context transition system *(SCTS for short) for a system $\mathcal{I} = \langle \mathbf{C}, \mathbb{X}, O, tr \rangle$ is a transition system $\beta \subseteq X \times ||\mathbf{C}|| \times O \times X$.*

In [12], we have introduced a SCTS for open nets. Intuitively the symbolic transition $N, m \xrightarrow{i,\lambda}_{\eta} N, m'$ is possible if and only if $N, m \oplus i \xrightarrow{\lambda}_{\mathcal{N}} N, m'$ and i is the smallest multiset (on input places) allowing such transition. This SCTS is formally defined by the following rule.

$$\frac{t \in T \quad l(t) = \lambda \quad m = (m \cap^{\bullet} t) \oplus n \quad i \subseteq I^{\oplus} \quad {}^{\bullet}t = (m \cap^{\bullet} t) \oplus i}{N, m \xrightarrow{i,\lambda}_{\eta} N, t^{\bullet} \oplus n}$$

The marking $m \cap^{\bullet} t$ contains all the tokens of m that are needed to perform the transition t. The marking n contains all the tokens of m that are not useful for performing t, while the marking i contains all the tokens that m needs to reach ${}^{\bullet}t$. Note that i is exactly the *smallest* multiset that is needed to perform the transition t. Indeed if we take i_1 strictly included into i, $m \oplus i_1$ cannot match ${}^{\bullet}t$. As an example consider the net N_2 in Fig.2 with marking $cd\$^2$ and let t be the only transition labeled with α. We have that $cd\$^2 \cap^{\bullet} t = c\$^2$, $n = d$ and $i = \$^3$. Thus $N_2, cd\$^2 \xrightarrow{\$^3,\alpha}_{\eta} N_2, dd$. Fig.2(C) shows symbolic transition systems of marked open nets discussed in the previous subsection.

Definition 5 (Inference System). *An inference system T for a context interactive system $\mathcal{I} = \langle \mathbf{C}, \mathbb{X}, O, tr \rangle$ is a set of rules of the following format, where $i, j \in |\mathbf{C}|$, $o, o' \in O$, $c \in \mathbf{C}[i, i']$ and $d \in \mathbf{C}[j, j']$.*

$$\frac{p_i \xrightarrow{o} q_j}{c(p_i) \xrightarrow{o'} d(q_j)}$$

The above rule states that all processes with interface i that perform a transition with observation o going into a state q_j with interface j, when inserted into the context c can perform a transition with the observation o' going into $d(q_j)$.

In the following, we write $c \xrightarrow[o']{o} d$ to mean a rule like the above. The rules $c \xrightarrow[o']{o} c'$ and $d \xrightarrow[o'']{o'} d'$ *derive* the rule $c; d \xrightarrow[o'']{o} c'; d'$ if $c; d$ and $c'; d'$ are defined. Given an inference system T, $\Phi(T)$ is the set of all the rules derivable from T together with the identities rules ($\forall o \in O$ and $\forall i, j \in |\mathbf{C}|$, $id_i \xrightarrow[o]{o} id_j$).

Definition 6 (Derivations, soundness and completeness). *Let $\mathcal{I}$ be a context interactive system, β an SCTS and T an inference system.*

We say that $p \xrightarrow{c_1,o_1} p_1$ derives $p \xrightarrow{c_2,o_2} p_2$ *in* T *(written* $p \xrightarrow{c_1,o_1} p_1 \vdash_T p \xrightarrow{c_2,o_2} p_2$*) if there exist* $d, e \in ||\mathbf{C}||$ *such that* $d \xrightarrow[o_2]{o_1} e \in \Phi(T)$*,* $c_1; d = c_2$ *and* $e_{\mathbb{X}}(p_1) = p_2$*.*

We say that β *and* T *are* sound and complete *w.r.t.* $\mathcal{I}$ *if*

$$p \xrightarrow{c,o}_S q \text{ iff } p \xrightarrow{c',o'}_\beta q' \text{ and } p \xrightarrow{c',o'}_\beta q' \vdash_T p \xrightarrow{c,o}_S q.$$

A sound and complete SCTS could be considerably smaller than the saturated transition system, but still containing all the information needed to recover $\sim^S$. Note that the ordinary bisimilarity over SCTS (hereafter called *syntactical bisimilarity* and denoted by $\sim^W$) is usually stricter than $\sim^S$. As an example consider the marked open nets a and l. These are not syntactically bisimilar, since $l \xrightarrow{\$^3,\alpha}_\eta m$ while a cannot (Fig.2(C)). However, they are saturated bisimilar, as discussed in the previous subsection. In order to recover $\sim^S$ through the symbolic transition system we need a more elaborated definition of bisimulation.

Definition 7 (Symbolic Bisimilarity). *Let* $\mathcal{I} = \langle \mathbf{C}, \mathbb{X}, O, tr \rangle$ *be an interactive system,* T *be a set of rules and* β *be a symbolic transition system. Let* $R = \{R_i \subseteq X_i \times X_i \mid i \in |\mathbf{C}|\}$ *be an* $|\mathbf{C}|$ *sorted family of symmetric relations.* R *is a* symbolic bisimulation *iff* $\forall i \in |\mathbf{C}|$*, whenever* pR_iq*:*

– *if* $p \xrightarrow{c,o}_\beta p'$*, then* $q \xrightarrow{c_1,o_1}_\beta q'_1$ *and* $q \xrightarrow{c_1,o_1}_\beta q'_1 \vdash_T q \xrightarrow{c,o} q'$ *and* $p' R_k q'$*.*

We write $p \sim_i^{SYM} q$ *iff there exists a symbolic bisimulation* R *such that* pR_iq*.*

Theorem 1. *Let* $\mathcal{I}$ *be a context interactive system,* β *an* SCTS *and* T *an inference system. If* β *and* T *are sound and complete w.r.t.* $\mathcal{I}$*, then* $\sim^{SYM} = \sim^S$*.*

In the remainder of this section we focus on open Petri nets. The inference system $T_\mathcal{N}$ is defined by the following parametric rule.

$$\frac{N, m \xrightarrow{\lambda}_\mathcal{N} N, m'}{N, m \oplus i \xrightarrow{\lambda}_\mathcal{N} N, m' \oplus i}$$

The intuitive meaning of this rule is that for all possible observations λ and multiset i on input places, if a marked net performs a transition with observation λ, then the addition of i preserves this transition.

Now, consider derivations between transitions of open nets. It is easy to see that $N, m \xrightarrow{i_1,\lambda_1} N, m_1 \vdash_{T_\mathcal{N}} N, m \xrightarrow{i_2,\lambda_2} N, m_2$ if and only if $\lambda_2 = \lambda_1$ and there exists a multiset x on input places such that $i_2 = i_1 \oplus x$ and $m_2 = m_1 \oplus x$. For all the nets N_k of our example, this just means that for all observations λ and for all multisets m, n, we have that $\langle N_k, m \rangle \xrightarrow{\$^i,\lambda}_\eta \langle N_k, n \rangle \vdash_{T_\mathcal{N}} \langle N_k, m \rangle \xrightarrow{\$^{i+j},\lambda} \langle N_k, n\$^j \rangle$.

In [12] we have shown that $T_\mathcal{N}$ and η are sound and complete w.r.t. $\mathcal{N}$.

3 (Structured) Coalgebras

In this section we recall the notions of the theory of coalgebras that will be useful in the following to give coalgebraic models for saturated and symbolic semantics.

Definition 8 (Coalgebra). *Let* $\mathbf{B} : \mathbf{C} \to \mathbf{C}$ *be an endofunctor on a category* $\mathbf{C}$*. A* $\mathbf{B}$-coalgebra *is a pair* $\langle X, \alpha \rangle$ *where* X *is an object of* $\mathbf{C}$ *and* $\alpha : X \to \mathbf{B}(X)$ *is an arrow. A* $\mathbf{B}$-cohomomorphism $f : \langle X, \alpha \rangle \to \langle Y, \beta \rangle$ *is an arrow* $f : X \to Y$ *of* $\mathbf{C}$ *such that* $f; \beta = \alpha; \mathbf{B}(f)$*.* $\mathbf{B}$*-coalgebras and* $\mathbf{B}$*-cohomomorphisms forms the category* $\mathbf{Coalg_B}$*.*

For example, the functor $\mathbf{P_L} : \mathbf{Set} \to \mathbf{Set}$ defined as $\mathbf{P}(L \times id)$ (for L a set of labels and $\mathbf{P}$ the powerset functor) defines the category $\mathbf{Coalg_{P_L}}$ of L-labeled transition systems and "zig-zag" morphisms.

If $\mathbf{Coalg_B}$ has a final object, one can define the behaviour of a $\mathbf{B}$-coalgebra as the final morphism. Thus behavioural equivalence, hereafter referred as bisimilarity, is defined as the kernel of a final morphism.Moreover, in a final coalgebra all bisimilar states are identified, and thus, the image of a coalgebra through a final morphism is its minimal realization (w.r.t. bisimilarity). In the finite case, this can be computed via a minimization algorithm.

Unfortunately, due to cardinality reasons, $\mathbf{Coalg_{P_L}}$ does not have a final object [1]. One satisfactory solution consists in replacing the powerset functor $\mathbf{P}$ by the *countable* powerset functor $\mathbf{P}_c$, which maps a set to the family of its countable subsets. Then, by defining $\mathbf{P^c_L} = \mathbf{P}_c(L \times id)$, one has that coalgebras for this endofunctor are one-to-one with transition systems with *countable degree.* Unlike functor $\mathbf{P_L}$, functor $\mathbf{P^c_L}$ admits final coalgebras (Ex. 6.8 of [1]).

The coalgebraic representation using functor $\mathbf{P^c_L}$ is not completely satisfactory, because the intrinsic algebraic structure of states is lost. This calls for the introduction of *structured coalgebras* [21], i.e. coalgebras for an endofuctor on a category $\mathbf{Alg_\Gamma}$ of algebras for a specification Γ. Since cohomomorphisms in a category of structured coalgebras are also Γ-homomorphisms, *bisimilarity* (i.e. the kernel of a final morphism) *is a congruence* w.r.t. the operations in Γ.

Moreover, since we would like that the structured coalgebraic model is compatible with the unstructured, set-based one, we are interested in functors $\mathbf{B^\Gamma} : \mathbf{Alg_\Gamma} \to \mathbf{Alg_\Gamma}$ that are *lifting* of some functor $\mathbf{B} : \mathbf{Set} \to \mathbf{Set}$ along the *forgetful* functor $\mathbf{V^\Gamma} : \mathbf{Alg_\Gamma} \to \mathbf{Set}$ (i.e., $\mathbf{B^\Gamma}; \mathbf{V^\Gamma} = \mathbf{V^\Gamma}; \mathbf{B}$).

Proposition 2 (From [21]). *Let* Γ *be an algebraic specification. Let* $\mathbf{V^\Gamma} : \mathbf{Alg_\Gamma} \to \mathbf{Set}$ *be the forgetful functor. If* $\mathbf{B_\Gamma} : \mathbf{Alg_\Gamma} \to \mathbf{Alg_\Gamma}$ *is a lifting of* $\mathbf{P^c_L}$ *along* $\mathbf{V^\Gamma}$*, then* (1) $\mathbf{Coalg_{B_\Gamma}}$ *has a final object,* (2) *bisimilarity is uniquely induced by bisimilarity in* $\mathbf{Coalg_{P^c_L}}$ *and* (3) *bisimilarity is a congruence.*

In [13], *bialgebras* are used as structures combining algebras and coalgebras. Bialgebras are richer than structured coalgebras, in the sense that they can be seen both as coalgebras on algebras and also as algebras on coalgebras. In Section 5.2, we will introduce *normalized coalgebras* that are not bialgebras. This explains why we decided to use structured coalgebras.

4 Coalgebraic Saturated Semantics

In this section we introduce the coalgebraic model for the saturated transition system. First we model it as a coalgebra over $\mathbf{Set}^{|\mathbf{C}|}$, i.e., the category of $|\mathbf{C}|$-sorted families of sets and functions. Therefore in this model, all the algebraic structure is missing. Then we lift it to $\mathbf{Alg}_{\mathbf{\Gamma(C)}}$ that is the category of $\Gamma(\mathbf{C})$-algebras and $\Gamma(\mathbf{C})$-homomorphisms.

In the following, we assume that the SATTS has countable degree.

Definition 9. $\mathbf{G} : \mathbf{Set}^{|\mathbf{C}|} \to \mathbf{Set}^{|\mathbf{C}|}$ *is defined for each* $|\mathbf{C}|$*-sorted family of set* X *and for each* $i \in |\mathbf{C}|$ *as* $\mathbf{G}(X_i) = \mathbf{P}_c(\sum_{j \in |\mathbf{C}|}(\mathbf{C}[i,j] \times O \times \sum_{k \in |\mathbf{C}|} X_k))$. *Analogously for arrows.*

A $\mathbf{G}$-coalgebra is a $\mathbf{C}$-sorted family $\alpha = \{\alpha_i : X_i \to \mathbf{G}(X_i) \mid i \in |\mathbf{C}|\}$ of functions assigning to each $p \in X_i$ a set of transitions (c, o, q) where c is an arrow of $\mathbf{C}$ (context) with source i, o is an observation and q is the arriving state. Note that q can have any possible sort ($q \in \sum_{k \in |\mathbf{C}|} X_k$).

Then, given $\mathcal{I} = \langle \mathbf{C}, \mathbb{X}, O, tr \rangle$, we define the $\mathbf{G}$-coalgebra $\langle X, \alpha_{\mathcal{I}} \rangle$ corresponding to the SATTS, where $\forall i \in |\mathbf{C}|$, $\forall p \in X_i$, $(c, o, q) \in \alpha_{\mathcal{I}}(p)$ iff $(c_{\mathbb{X}}(p), o, q) \in tr$.

Now we want to define an endofunctor $\mathbf{H}$ on $\mathbf{Alg}_{\mathbf{\Gamma(C)}}$ that is a lifting of $\mathbf{G}$ and such that $\langle \mathbb{X}, \alpha_{\mathcal{I}} \rangle$ is a $\mathbf{H}$-coalgebra. In order to do that, we must define how $\mathbf{H}$ modifies the operations of $\Gamma(\mathbf{C})$-algebras. Usually, this is done by giving a set of GSOS rules.

In our case, the following (parametric) rule defines $\mathbf{H}$.

$$\frac{p \xrightarrow{c_1, l} q \quad c_1 = d; c_2}{d(p) \xrightarrow{c_2, l} q}$$

Hereafter, in order to make lighter the notation, we will avoid to specify sorts. We will denote a $\Gamma(\mathbf{C})$-algebra $\mathbb{X}$ as $\langle X, d^0_{\mathbb{X}}, d^1_{\mathbb{X}}, \dots \rangle$ where X is the $|\mathbf{C}|$-sorted carrier set of $\mathbb{X}$ and $d^i_{\mathbb{X}}$ is the function corresponding to the operator $d^i \in ||\mathbf{C}||$.

Definition 10. $\mathbf{H} : \mathbf{Alg}_{\mathbf{\Gamma(C)}} \to \mathbf{Alg}_{\mathbf{\Gamma(C)}}$ *maps each* $\mathbb{X} = \langle X, d^0_{\mathbb{X}}, d^1_{\mathbb{X}}, \dots \rangle \in \mathbf{Alg}_{\mathbf{\Gamma(C)}}$ *into* $\langle \mathbf{G}(X), d^0_{\mathbf{H}(\mathbb{X})}, d^1_{\mathbf{H}(\mathbb{X})}, \dots \rangle$ *where* $\forall d \in \Gamma(\mathbf{C})$, $\forall A \in \mathbf{G}(X)$, $d_{\mathbf{H}(\mathbb{X})} A = \{(c_2, l, x)$ *s.t.* $(c_1, l, x) \in A$ *and* $c_1 = d; c_2\}$. *For arrows, it is defined as* $\mathbf{G}$.

It is immediate to see that $\mathbf{H}$ is a lifting of $\mathbf{G}$. Thus, by Prop.2, follows that $\mathbf{Coalg_H}$ has final object and that bisimilarity is a congruence.[2] In the following, we proved that $\alpha_{\mathcal{I}} : \mathbb{X} \to \mathbf{H}(\mathbb{X})$ is a $\Gamma(\mathbf{C})$-homomorphism.

Theorem 2. $\langle \mathbb{X}, \alpha_{\mathcal{I}} \rangle$ *is a* $\mathbf{H}$*-coalgebra.*

Now, since a final coalgebra $F_{\mathbf{H}}$ exists in $\mathbf{Coalg_H}$ and since $\langle \mathbb{X}, \alpha_{\mathcal{I}} \rangle$ is a $\mathbf{H}$-coalgebra, then the kernel of its final morphism coincides with $\sim^S$.

[2] Prop. 2 holds also for many-sorted algebras and many sorted sets [22].

5 Coalgebraic Symbolic Semantics

In Sec. 4 we have characterized saturated bisimilarity as the equivalence induced by a final morphisms from $\langle \mathbb{X}, \alpha_{\mathcal{I}} \rangle$ (i.e., the $\mathbf{H}$-coalgebra corresponding to SATTS) to $F_{\mathbf{H}}$. This is theoretically interesting, but pragmatically useless. Indeed SATTS is usually infinite branching (or in any case very inefficient), and so it is the minimal model. In this section we use symbolic bisimilarity in order to give an efficient and coalgebraic characterization of $\sim^S$. We provide a notion of *redundant transitions* and we introduce *normalized coalgebras* as coalgebras without redundant transitions. The category of normalized coalgebras ($\mathbf{Coalg_{N_T}}$) is isomorphic to the category of *saturated coalgebras*, i.e., the full subcategory of $\mathbf{Coalg_H}$ that contains only those coalgebras "satisfying" an inference system T. From the isomorphism follows that $\sim^S$ coincides with the kernel of the final morphism in $\mathbf{Coalg_{N_T}}$. This provides a characterization of $\sim^S$ really useful: every equivalence class has a canonical model that is smaller than that in $\mathbf{Coalg_H}$ because normalized coalgebras have not redundant transitions. Moreover, minimizing in $\mathbf{Coalg_{N_T}}$ is feasible since it abstracts away from redundant transitions.

5.1 Saturated Coalgebras

Hereafter we refer to a context interactive system $\mathcal{I} = \langle \mathbf{C}, \mathbb{X}, O, tr \rangle$ and to an inference system T. First, we extend $\vdash_T$ (Def.6) to $\Gamma(\mathbf{C})$-algebras.

Definition 11 (extended derivation). *Let $\mathbb{X}$ be a $\Gamma(\mathbf{C})$-algebra. A transition $p \xrightarrow{c_1, o_1} q_1$* derives *a transition $d_{\mathbb{X}}(p) \xrightarrow{c_2, o_2} q_2$ in $\mathbb{X}$ through T (written $(c_1, o_1, q_1) \vdash^d_{T,\mathbb{X}} (c_2, o_2, q_2)$) iff there exists $e \in ||\mathbf{C}||$ such that $c_1; e = d; c_2$ and $\phi : o_1 \xrightarrow[e']{e} o_2 \in \Phi(T)$ and $e'_{\mathbb{X}}(q_1) = q_2$.*

Intuitively, $\vdash^d_{T,\mathbb{X}}$ allows to derive from the set of transitions of a state p some transitions of $d_{\mathbb{X}}(p)$. Consider the symbolic transition $p \xrightarrow{\$^3, \alpha}_\eta m$ of l (Fig.2) in our running example. The derivation $(\$^3, \alpha, m) \vdash^{\$^2}_{T_{\mathcal{N}},\mathbb{N}} (\$, \alpha, m) \vdash^{\$^2}_{T_{\mathcal{N}},\mathbb{N}} (\$, \alpha, m\$^2)$ means that $l\$^2 \xrightarrow{\$, \alpha} m$ and $l\$^4 \xrightarrow{\$, \alpha} m\$^2$. The latter is not a symbolic transition.

Definition 12 (Sound Inference System). *A inference system T is* sound *w.r.t. a $\mathbf{H}$-coalgebra $\langle \mathbb{X}, \alpha \rangle$ (or viceversa, $\langle \mathbb{X}, \alpha \rangle$* satisfies *$T$) iff whenever $(c, o, q) \in \alpha(p)$ and $(c, o, q) \vdash^d_{T,\mathbb{X}} (c', o', q')$ then $(c', o', q') \in \alpha(d_{\mathbb{X}}(p))$.*

For example, $\langle \mathbb{N}, \alpha_{\mathcal{N}} \rangle$ (i.e., the $\mathbf{H}$-coalgebra corresponding to the SATTS of our running example) satisfies $T_{\mathcal{N}}$, while the coalgebra corresponding to the symbolic transition systems does not. Hereafter we use $\vdash_{T,\mathbb{X}}$ to mean $\vdash^{id}_{T,\mathbb{X}}$.

Definition 13 (Saturated set). *Let $\mathbb{X}$ be a $\Gamma(\mathbf{C})$-algebra. A set $A \in \mathbf{G}(X)$ is* saturated *in T and $\mathbb{X}$ if it is closed w.r.t. $\vdash_{T,\mathbb{X}}$. The set $\mathbf{S}^{\mathbf{T}\mathbb{Y}}(X)$ is the subset of $\mathbf{G}(X)$ containing all and only the saturated sets in T and $\mathbb{X}$.*

Definition 14. $\mathbf{S_T} : \mathbf{Alg_{\Gamma(C)}} \to \mathbf{Alg_{\Gamma(C)}}$ *maps each $\mathbb{X} = \langle X, d^0_{\mathbb{X}}, d^1_{\mathbb{X}}, \dots \rangle \in$* $\mathbf{Alg_{\Gamma(C)}}$ *into $\mathbf{S_T}(\mathbb{X}) = \langle \mathbf{S}^{\mathbf{T}\mathbb{X}}(X), d^0_{\mathbf{S_T}(\mathbb{X})}, d^1_{\mathbf{S_T}(\mathbb{X})}, \dots \rangle$ where $\forall d \in \Gamma(\mathbf{C})$, $\forall A \in$*

$\mathbf{G}(X)$, $d_{\mathbf{S_T}(\mathbb{X})}A = \{(c_2, o_2, x_2)$ *s.t.* $(c_1, o_1, x_1) \in A$ *and* $(c_1, o_1, x_1) \vdash^d_{T,\mathbb{X}} (c_2, o_2, x_2)\}$. *For arrows, it is defined as* $\mathbf{G}$.

There are two differences w.r.t $\mathbf{H}$. First, we require that all the sets of transitions are saturated. Then the operators are defined by using the relation $\vdash^d_{T,\mathbb{X}}$.

Notice that $\mathbf{S_T}$ cannot be regarded as a lifting of any endofunctor over $\mathbf{Set}^{|\mathbf{C}|}$. Indeed the definition of $\mathbf{S}^{\mathbf{T}\mathbb{X}}(X)$ depends from the algebraic structure $\mathbb{X}$. For this reason we cannot use Prop.2. In order to prove the existence of final object in $\mathbf{Coalg_{S_T}}$, we show that $\mathbf{Coalg_{S_T}}$ is the full subcategory of $\mathbf{Coalg_H}$ containing all and only the coalgebras satisfying T. More precisely, we show that $|\mathbf{Coalg_{S_T}}|$ is a *covariety* of $\mathbf{Coalg_H}$.

Lemma 1. *A* $\mathbf{H}$*-coalgebra is a* $\mathbf{S_T}$*-coalgebra iff satisfies* T.

Proposition 3. $|\mathbf{Coalg_{S_T}}|$ *is a covariety of* $\mathbf{Coalg_H}$.

From this follows that we can construct a final object in $\mathbf{Coalg_{S_T}}$ as the biggest subobject of $F_{\mathbf{H}}$ satisfying T. Thus, the kernel of final morphism from $\langle \mathbb{X}, \alpha_{\mathcal{I}} \rangle$ in $\mathbf{Coalg_{S_T}}$ coincides with $\sim^S$.

5.2 Normalized Coalgebras

In this subsection we introduce normalized coalgebras, in order to characterize $\sim^S$ without considering the whole SATTS and by relying on the derivation relation $\vdash_{T,\mathbb{X}}$. The following observation is fundamental to explain our idea.

Lemma 2. *Let* $\mathbb{X}$ *be a* $\Gamma(\mathbf{C})$*-algebra. If* $(c_1, o_1, p_1) \vdash_{T,\mathbb{X}} (c_2, o_2, p_2)$ *then* $p_2 = e_{\mathbb{X}}(p_1)$ *for some* $e \in ||\mathbf{C}||$. *Moreover* $\forall q_1 \in X$, $(c_1, o_1, q_1) \vdash_{T,\mathbb{X}} (c_2, o_2, e_{\mathbb{X}}(q_1))$.

Consider a $\mathbf{H}$-coalgebra $\langle \mathbb{X}, \gamma \rangle$ and the equivalence $\sim^\gamma$ induced by the final morphism. Suppose that $p \xrightarrow{c_1, o_1}_\gamma p_1$ and $p \xrightarrow{c_2, o_2}_\gamma p_2$ such that $(c_1, o_1, p_1) \vdash_{T,\mathbb{X}} (c_2, o_2, e_{\mathbb{X}}(p_1))$. If $\langle \mathbb{X}, \gamma \rangle$ satisfies T (i.e., it is a $\mathbf{S_T}$-coalgebra), we can forget about the latter transition. Indeed, for all $q \in X$, if $q \xrightarrow{c_1, o_1}_\gamma q_1$ then also $q \xrightarrow{c_2, o_2}_\gamma e_{\mathbb{X}}(q_1)$ (since $\langle \mathbb{X}, \gamma \rangle$ satisfies T) and if $p_1 \sim^\gamma q_1$, then also $e_{\mathbb{X}}(p_1) \sim^\gamma e_{\mathbb{X}}(q_1)$ (since $\sim^\gamma$ is a congruence). Thus, when checking bisimilarity, we can avoid to consider those transitions that are derivable from others. We call such transitions *redundant*.

The wrong way to efficiently characterize $\sim^\gamma$ by exploiting $\vdash_{T,\mathbb{X}}$, consists in removing all the redundant transitions from $\langle \mathbb{X}, \gamma \rangle$ obtaining a new coalgebra $\langle \mathbb{X}, \beta \rangle$ and then computing $\sim^\beta$ (i.e., the ordinary bisimilarity on $\langle \mathbb{X}, \beta \rangle$). When considering $\langle \mathbb{X}, \alpha_{\mathcal{I}} \rangle$ (i.e., the $\mathbf{H}$-coalgebra corresponding to SATTS), this roughly means to build a symbolic transition system and then computing the ordinary bisimilarity over this. But, as we have seen in Sec.2, the resulting bisimilarity (denoted by $\sim^W$) does not coincide with the original one.

Generally, this happens when $p \xrightarrow{c_1, o_1}_\beta p_1$ and $p \xrightarrow{c_2, o_2}_\beta p_2$ with $(c_1, o_1, p_1) \vdash_{T,\mathbb{X}} (c_2, o_2, e_{\mathbb{X}}(p_1))$ and $e_{\mathbb{X}}(p_1) \neq p_2$, but $e_{\mathbb{X}}(p_1) \sim^\gamma p_2$. Notice that the latter transitions is not removed, because it is not considered redundant since $e_{\mathbb{X}}(p_1)$ is different from p_2 (even if semantically equivalent). A transition as the latter is called *semantically redundant* and it causes the mismatch between $\sim^\beta$ and $\sim^\gamma$.

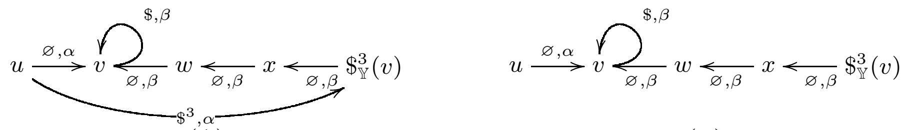


Fig. 3. (A) Part of a not normalized coalgebra $\langle \mathbb{Y}, \gamma \rangle$. (B) Part of a normalized coalgebra $\langle \mathbb{Y}, \gamma; norm_{\mathbb{Y},T_{\mathcal{N}}} \rangle$.

Indeed, take a process q performing $q \xrightarrow{c_1,o_1}_\beta q_1$ with $p_1 \sim^\gamma q_1$. Clearly $p \not\sim^\beta q$, but $p \sim^\gamma q$. Indeed $(c_1, o_1, q_1) \vdash_{T,\mathbb{X}} (c_2, o_2, e_{\mathbb{X}}(q_1))$ and thus $q \xrightarrow{c_2,o_2}_\gamma e_{\mathbb{X}}(q_1)$ (since $\langle \mathbb{X}, \gamma \rangle$ satisfies T) and $p_2 \sim^\gamma e_{\mathbb{X}}(p_1) \sim^\gamma e_{\mathbb{X}}(q_1)$ (since $\sim^\gamma$ is a congruence).

As an example consider the symbolic transition system of l (Fig.2). $l \xrightarrow{\varnothing,\alpha}_\eta q$ and $l \xrightarrow{\$^3,\alpha}_\eta m$. Moreover, $l \xrightarrow{\varnothing,\alpha}_\eta q \vdash_{T_{\mathcal{N}},\mathbb{N}} l \xrightarrow{\$^3,\alpha} \$^3_{\mathbb{N}}(q)$ and $\$^3_{\mathbb{N}}(q) = q\$^3 \neq m$, but $q\$^3 \sim^S m$. Now consider a. $a \xrightarrow{\varnothing,\alpha}_\eta b$. Clearly $l \not\sim^W a$ but $l \sim^S a$ (Sec. 2).

The above observation tells us that we have to remove not only the redundant transition, i.e., those derivable from $\vdash_{T,\mathbb{X}}$, but also the *semantically redundant* ones. But immediately a problem arises. How can we decide which transitions are semantically redundant, if semantic redundancy itself depends on bisimilarity?

Our solution is the following: we define a category of coalgebras without redundant transitions ($\mathbf{Coalg}_{\mathbf{N_T}}$) and, as a result, a final coalgebra contains no semantically redundant transitions.

Definition 15 (Normalized Set and Normalization). *Let $\mathbb{X}$ be a $\Gamma(\mathbf{C})$-algebra. A transition (c', o', q') is* equivalent *to (c, o, q) in $T, \mathbb{X}$ (written $(c', o', q') \equiv_{T,\mathbb{X}} (c, o, q)$) iff $(c', o', q') \vdash_{T,\mathbb{X}} (c, o, q)$ and $(c, o, q) \vdash_{T,\mathbb{X}} (c', o', q')$. A transition (c', o', q')* dominates *(c, o, q) in $T, \mathbb{X}$ (written $(c', o', q') \prec_{T,\mathbb{X}} (c, o, q)$) iff $(c', o', q') \vdash_{T,\mathbb{X}} (c, o, q)$ and $(c, o, q) \nvdash_{T,\mathbb{X}} (c', o', q')$. Let and $A \in \mathbf{G}(Y)$. A transition $(c, o, q) \in A$ is* redundant *in A w.r.t. $T, \mathbb{X}$ if $\exists (c', o', q') \in A$ such that $(c', o', q') \prec_{T,\mathbb{X}} (c, o, q)$. The set A is* normalized *w.r.t. $T, \mathbb{X}$ iff it does not contain redundant transitions and it is closed by equivalent transitions. The set $\mathbf{N}^{\mathbf{T}}_{\mathbb{X}}(Y)$ is the subset of $\mathbf{G}(Y)$ containing all and only the normalized sets w.r.t. $T, \mathbb{X}$. The* normalization function *$norm_{T,\mathbb{Y}} : \mathbf{G}(Y) \to \mathbf{N}^{\mathbf{T}}_{\mathbb{X}}(Y)$ maps $A \in \mathbf{G}(Y)$ into $\{(c', o', q') s.t.\ \exists (c, o, q) \in A$ s.t. $(c', o', q') \equiv_{T,\mathbb{X}} (c, o, q)$ and (c, o, q) not redundant in A w.r.t. $T, \mathbb{X}\}$.*

Recall $\mathcal{N} = \langle \mathbf{Tok}, \mathbb{N}, \Lambda, tr_{\mathcal{N}} \rangle$ and $T_{\mathcal{N}}$. Consider the coalgebra $\langle \mathbb{Y}, \gamma \rangle$ partially depicted in Fig.3(A). Here we have that $(\varnothing, \alpha, v) \vdash_{T_{\mathcal{N}},\mathbb{Y}} (\$^3, \alpha, \$^3_{\mathbb{Y}}(v))$ but $(\$^3, \alpha, \$^3_{\mathbb{Y}}(v)) \nvdash_{T_{\mathcal{N}},\mathbb{Y}} (\varnothing, \alpha, v)$. Thus the set $\gamma(u)$, i.e., the set of transitions of u, is not normalized (w.r.t $T_{\mathcal{N}}, \mathbb{Y}$) since the transition $(\$^3, \alpha, \$^3_{\mathbb{Y}}(v))$ is redundant in $\gamma(u)$ (it is dominated by $(\varnothing, \alpha, v)$). By applying $norm_{\mathbb{Y},T_{\mathcal{N}}}$ to $\gamma(u)$, we get the normalized set of transitions $\{(\varnothing, \alpha, v)\}$ (Fig.3(B)). It is worth noting that in open petri nets, two transitions are equivalent iff they are the same transition.

Definition 16. $\mathbf{N_T} : \mathbf{Alg_{\Gamma(C)}} \to \mathbf{Alg_{\Gamma(C)}}$ *maps each* $\mathbb{X} = \langle X, d^1_{\mathbb{X}}, d^2_{\mathbb{X}}, \dots \rangle \in \mathbf{Alg_{\Gamma(C)}}$ *into* $\mathbf{N_T}(\mathbb{X}) = \langle \mathbf{N}^{\mathbf{T}}_{\mathbb{X}}(X), d^1_{\mathbf{S_T}(\mathbb{X})}; norm_{T,\mathbb{X}}, d^2_{\mathbf{S_T}(\mathbb{X})}; norm_{T,\mathbb{X}}, \dots \rangle$*. For all* $h : \mathbb{X} \to \mathbb{Y}$*,* $\mathbf{N_T}(h) = \mathbf{H}(h); norm_{T,\mathbb{Y}}$*.*

Thus, the coalgebra $\langle \mathbb{Y}, \gamma \rangle$ partially represented in Fig.3(A) is not normalized, while $\langle \mathbb{Y}, \gamma; norm_{T,\mathbb{Y}} \rangle$ in Fig.3(B) is. In order to get a normalized coalgebra for our running example, we can normalize its saturated coalgebra $\langle \mathbb{N}, \alpha_{\mathcal{N}} \rangle$ obtaining $\langle \mathbb{N}, \alpha_{\mathcal{N}}; norm_{T_{\mathcal{N}},\mathbb{N}} \rangle$. For the nets shown in Fig.2, this coincides with the SCTS η.

The most important idea behind normalized coalgebra is in the definition of $\mathbf{N_T}(h)$: we first apply $\mathbf{H}(h)$ and then the normalization $norm_{T,\mathbb{Y}}$. Thus $\mathbf{N_T}$-cohomomorphisms must preserve not all the transitions of the source coalgebras, but only those that are not redundant when mapped into the target.

As an example consider the coalgebras $\langle \mathbb{N}, \alpha_{\mathcal{N}}; norm_{T_{\mathcal{N}},\mathbb{N}} \rangle$. For the state l, it coincides with the SCTS η (Fig. 2(C)). Consider the coalgebra $\langle \mathbb{Y}, \gamma; norm_{T_{\mathcal{N}},\mathbb{Y}} \rangle$ (partially represented in Fig.3(B)) and the $\Gamma(\mathbf{Tok})$-homomorphism $h : \mathbb{N} \to \mathbb{Y}$ that maps l, m, n, o into $u, \$^3_{\mathbb{Y}}(v), x, w$ (respectively) and p, q into v. Note that the transition $l \xrightarrow{\$^3,\alpha}_{\eta} m$ is not preserved (i.e., $u \not\xrightarrow{\$^3,\alpha}_{\gamma} h(m)$), but h is however a $\mathbf{N_T}$-cohomomorphism, because the transition $(\$^3, \alpha, h(m)) \in \mathbf{H}(h)(\eta(l))$ is removed by $norm_{T_{\mathcal{N}},\mathbb{Y}}$. Indeed $h(m) = \$^3_{\mathbb{Y}}(v)$ and $(\varnothing, \alpha, v) \vdash_{T_{\mathcal{N}},\mathbb{Y}} (\$^3, \alpha, \$^3_{\mathbb{Y}}(v))$. Thus, we forget about $l \xrightarrow{\$^3,\alpha}_{\eta} m$ that is, indeed, semantically redundant.

5.3 Isomorphism Theorem

Now we prove that $\mathbf{Coalg_{N_T}}$ is isomorphic to $\mathbf{Coalg_{S_T}}$.

Definition 17 (Saturation). *The function* $sat_{T,\mathbb{X}} : \mathbf{H}(X) \to \mathbf{S}^{\mathbf{T}\mathbb{X}}(X)$ *maps all* $A \in \mathbf{H}(X)$ *in* $\{(c', o', x') \text{ s.t. } (c, o, x) \in A \text{ and } (c, o, x) \vdash_{T,\mathbb{X}} (c', o', x')\}$*.*

Saturation is intuitively the opposite of normalization. Indeed saturation adds to a set all the redundant transitions, while normalization junks all of them. Thus, if we take a saturated set of transitions, we first normalize it and then we saturate it, we obtain the original set. Analogously for a normalized set.

However, in order to get such correspondence, we must add a constraint to our theory. Indeed, according to the actual definitions, there could exist a $\mathbf{S_T}$-coalgebra $\langle \mathbb{X}, \gamma \rangle$ and an infinite descending chain like: $\dots \prec_{T,\mathbb{X}} p \xrightarrow{c_2,o_2}_{\gamma} p_2 \prec_{T,\mathbb{X}} p \xrightarrow{c_1,o_1}_{\gamma} p_1$. In this chain, all the transitions are redundant and thus if we normalize it, we obtain an empty set of transitions.

Definition 18 (Normalizable System). *A context interactive system* $\mathcal{I} = \langle \mathbf{C}, \mathbb{X}, O, tr \rangle$ *is* normalizable w.r.t. T *iff* $\forall \mathbb{X} \in \mathbf{Alg_{\Gamma(C)}}$*,* $\prec_{T,\mathbb{X}}$ *is well founded.*

Lemma 3. *Let* $\mathcal{I}$ *be a normalizable system w.r.t.* T*. Let* $\mathbb{X}$ *be* $\Gamma(\mathbf{C})$*-algebra and* $A \in \mathbf{G}(Y)$*. Then* $\forall (d, o, x) \in A$*,* $\exists (d', o', x') \in norm_{T,\mathbb{Y}}(A)$*, such that* $(d', o', x') \prec_{T,\mathbb{X}} (d, o, x)$*.*

All the examples of [12] are normalizable [23]. Hereafter, we always refer to normalizable systems.

Proposition 4. *Let* $norm_T$ *and* sat_T *be respectively the families of morphisms* $\{norm_{T,\mathbb{X}} : \mathbf{S_T}(\mathbb{X}) \to \mathbf{N_T}(\mathbb{X}),\ \forall \mathbb{X} \in |\mathbf{Alg}_{\mathbf{\Gamma(C)}}|\}$ *and* $\{sat_{T,\mathbb{X}} : \mathbf{N_T}(\mathbb{X}) \to \mathbf{S_T}(\mathbb{X}),\ \forall \mathbb{X} \in |\mathbf{Alg}_{\mathbf{\Gamma(C)}}|\}$*. Then* $norm_T : \mathbf{S_T} \Rightarrow \mathbf{N_T}$ *and* $sat_T : \mathbf{N_T} \Rightarrow \mathbf{S_T}$ *are natural transformations. More precisely, they are natural isomorphisms, one the inverse of the other.*

It is well-known that any natural transformation between endofunctors induces a functor between the corresponding categories of coalgebras [1]. In our case, $norm_T : \mathbf{S_T} \Rightarrow \mathbf{N_T}$ induces the functor $\mathbf{NORM_T} : \mathbf{Coalg}_{\mathbf{S_T}} \to \mathbf{Coalg}_{\mathbf{N_T}}$ that maps every coalgebra $\langle \mathbb{X}, \alpha \rangle$ in $\langle \mathbb{X}, \alpha; norm_{T,\mathbb{X}} \rangle$ and every cohomomorphism h in itself. Analogously for sat_T.

Theorem 3. $\mathbf{Coalg}_{\mathbf{S_T}}$ *and* $\mathbf{Coalg}_{\mathbf{N_T}}$ *are isomorphic.*

Thus $\mathbf{Coalg}_{\mathbf{N_T}}$ has a final coalgebra $F_{\mathbf{N_T}}$ and the final morphisms from $\langle \mathbb{X}, \alpha_{\mathcal{I}}; norm_{T,\mathbb{X}} \rangle$ (that is $\mathbf{NORM_T}\langle \mathbb{X}, \alpha_{\mathcal{I}} \rangle$)) still characterizes $\sim^S$. This is theoretically very interesting, since the minimal canonical representatives of $\sim^S$ in $\mathbf{Coalg}_{\mathbf{N_T}}$ do not contain any (semantically) redundant transitions and thus they are much smaller than the (possibly infinite) minimal representatives in $\mathbf{Coalg}_{\mathbf{S_T}}$. Pragmatically, it allows for an effective procedure for minimizing that we will discuss in the next section. Notice that minimization is usually unfeasible in $\mathbf{Coalg}_{\mathbf{S_T}}$.

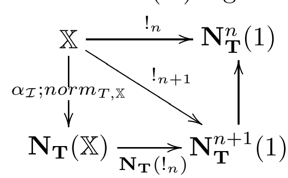
6 From Normalized Coalgebras to Symbolic Minimization

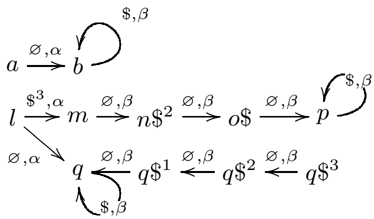
In [15], we have introduced a partition refinement algorithm for symbolic bisimilarity. First, it creates a partition P_0 equating all the states (with the same interface) of a symbolic transition system β and then, iteratively, refines this partition by splitting non equivalent states. The algorithm terminates whenever two subsequent partitions are equivalent. It computes the partition P_{n+1} as follows: p and q are equivalent in P_{n+1} iff whenever $p \xrightarrow{c,o}_\beta p_1$ is *not-redundant in* P_n, then $q \xrightarrow{c,o}_\beta q_1$ is *not-redundant in* P_n and p_1, q_1 are equivalent in P_n (and viceversa). By "not-redundant in P_n", we mean that no transition $p \xrightarrow{c',o'}_\beta p'_1$ exists such that $(c', o', p'_1) \vdash_{T,\mathbb{X}} (c, o, p'_2)$ and p'_2, p'_1 are equivalent in P_n.

Fig. 4 shows the partitions computed by the algorithm for the SCTS η of the marked nets a and l. Notice that a and l are equivalent in the partition P_1, because the transition $l \xrightarrow{\$^3,\alpha}_\eta m$ is redundant in P_0. Indeed, $l \xrightarrow{\varnothing,\alpha}_\eta q$, $(\varnothing, \alpha, q) \vdash_{T_\mathcal{N},\mathbb{N}} (\$^3, \alpha, q\$^3)$ and m is equivalent to $q\$^3$ in P_0.

Hereafter, we show that this algorithm corresponds to computing *the approximations* of a final morphism from $\langle \mathbb{X}, \alpha_{\mathcal{I}}; norm_{T,\mathbb{X}} \rangle$ to $F_{\mathbf{N_T}}$, as induced by the terminal sequence $1 \leftarrow \mathbf{N_T}(1) \leftarrow \mathbf{N}^2_\mathbf{T}(1) \leftarrow \dots$ where 1 is a final $\Gamma(\mathbf{C})$-algebra.

The $n+1$ approximation $!_{n+1}$ is defined as $\alpha_{\mathcal{I}}; norm_{T,\mathbb{X}}; \mathbf{N_T}(!_n)$. It is worth noting that $\forall n$ the kernel of $!_n$ coincides with the partition P_n computed by the algorithm. Indeed, we can safely replace $\alpha_{\mathcal{I}}; norm_{T,\mathbb{X}}$ with the SCTS β.





$$P_0 = \{a, l, b, p, q, q\$^1, o\$, q\$^2, n, q\$^3, m\}$$
$$P_1 = \{a, l\}\{b, p, q\}\{q\$^1, o\$, q\$^2, n, q\$^3, m\}$$
$$P_2 = \{a, l\}\{b, p, q\}\{q\$^1, o\$\}\{q\$^2, n, q\$^3, m\}$$
$$P_3 = \{a, l\}\{b, p, q\}\{q\$^1, o\$\}\{q\$^2, n\}\{q\$^3, m\}$$
$$P_4 = \{a, l\}\{b, p, q\}\{q\$^1, o\$\}\{q\$^2, n\}\{q\$^3, m\}$$

Fig. 4. The partitions computed for the marked nets a and l

Then, $!_{n+1} = \beta; \mathbf{N_T}(!_n)$. By the peculiar definition of $\mathbf{N_T}$ on arrows, $\mathbf{N_T}(!_n) = \mathbf{H}(!_n); norm_{T,\mathbf{N_T^n}(1)}$ and the normalization $norm_{T,\mathbf{N_T^n}(1)}$ exactly removes all the transitions that are redundant in P_n.

We end up this section by showing why we have used structured coalgebras instead of bialgebras. Bialgebras abstract away from the algebraic structure, while this is employed by the minimization in $\mathbf{Coalg_{N_T}}$. Indeed, in Fig.4, in order to compute the partitions of l, the algorithm needs the state $q\$^3$ that is not reachable from l. This happens because the algorithm must check if $l \xrightarrow{\$^3,\alpha}_\eta m$ is redundant. In [15], we have shown that we do not need the whole algebra but just a part of it, that can be computed in the initialization of the algorithm.

7 Conclusions and Related Works

The paper introduces two coalgebraic models for context interactive systems [12]. In the first one, the saturated transition system is an ordinary structured coalgebra $\langle \mathbb{X}, \alpha_{\mathcal{I}} \rangle$ and its final morphism induces $\sim^S$. The second model is the normalized coalgebra $\langle \mathbb{X}, \alpha_{\mathcal{I}}; norm_{T,\mathbb{X}} \rangle$ that is obtained by pruning all the redundant transitions from the first one. The equivalence induced by its final morphism is still $\sim^S$, but this characterization is much more convenient. Indeed, in the final normalized coalgebra all the (semantically) redundant transitions are eliminated. Moreover, minimization is feasible with normalized coalgebras and coincides with the symbolic minimization algorithm introduced in [15].

In [24], we have used normalized coalgebras for *Leifer and Milner's reactive systems* [25]. These are an instance of our contexts interactive systems (as shown in [12]) and thus the normalized coalgebras of [24] are a just special case of the one presented in this paper. More precisely, the coalgebras in [24] are defined for a special inference system, while those presented here are parametric w.r.t. it. This provides a flexible theory that gives coalgebraic semantics to many formalisms, such as mobile ambients [9], open [6] and asynchronous [8] π-calculus.

A coalgebraic model for mobile ambients has been proposed in [26]. However it characterizes *action bisimilarity* that is strictly included into *reduction barbed congruence* [10]. In [27], the authors show a context interactive system for mobile ambients, where the symbolic bisimilarity coincides with [10].

For asynchronous and open π-calculus, a minimization algorithm has been proposed in [28] and [29], respectively. In [15], we showed that these are special cases of our algorithm.

References

1. Rutten, J.: Universal coalgebra: a theory of systems. TCS 249(1), 3–80 (2000)
2. Kanellakis, P.C., Smolka, S.A.: Ccs expressions, finite state processes, and three problems of equivalence. Information and Computation 86(1), 43–68 (1990)
3. Milner, R.: Communicating and Mobile Systems: the π-Calculus. Cambridge University Press, Cambridge (1999)
4. Montanari, U., Sassone, V.: Dynamic congruence vs. progressing bisimulation for ccs. Fundam. Inform. 16(1), 171–199 (1992)
5. Milner, R., Parrow, J., Walker, D.: A calculus of mobile processes, i and ii. Information and Computation 100(1), 1–40, 41–77 (1992)
6. Sangiorgi, D.: A theory of bisimulation for the pi-calculus. Acta Inf. 33(1), 69–97 (1996)
7. Amadio, R.M., Castellani, I., Sangiorgi, D.: On bisimulations for the asynchronous pi-calculus. In: Sassone, V., Montanari, U. (eds.) CONCUR 1996. LNCS, vol. 1119, pp. 147–162. Springer, Heidelberg (1996)
8. Honda, K., Tokoro, M.: An object calculus for asynchronous communication. In: America, P. (ed.) ECOOP 1991. LNCS, vol. 512, pp. 133–147. Springer, Heidelberg (1991)
9. Cardelli, L., Gordon, A.D.: Mobile ambients. TCS 240(1), 177–213 (2000)
10. Merro, M., Nardelli, F.Z.: Bisimulation proof methods for mobile ambients. In: Baeten, J.C.M., Lenstra, J.K., Parrow, J., Woeginger, G.J. (eds.) ICALP 2003. LNCS, vol. 2719, pp. 584–598. Springer, Heidelberg (2003)
11. Hennessy, M., Lin, H.: Symbolic bisimulations. TCS 138(2), 353–389 (1995)
12. Bonchi, F., Montanari, U.: Symbolic semantics revisited. In: Amadio, R.M. (ed.) FOSSACS 2008. LNCS, vol. 4962, pp. 395–412. Springer, Heidelberg (2008)
13. Turi, D., Plotkin, G.D.: Towards a mathematical operational semantics. In: Proc. of LICS, pp. 280–291. IEEE, Los Alamitos (1997)
14. Corradini, A., Große-Rhode, M., Heckel, R.: Structured transition systems as lax coalgebras. ENTCS 11 (1998)
15. Bonchi, F., Montanari, U.: Minimization algorithm for symbolic bisimilarity. In: Proc. of ESOP. LNCS, vol. 5502, pp. 267–284. Springer, Heidelberg (2009)
16. Kindler, E.: A compositional partial order semantics for petri net components. In: Azéma, P., Balbo, G. (eds.) ICATPN 1997. LNCS, vol. 1248, pp. 235–252. Springer, Heidelberg (1997)
17. Parrow, J., Victor, B.: The fusion calculus: Expressiveness and symmetry in mobile processes. In: Proc. of LICS, pp. 176–185 (1998)
18. Wischik, L., Gardner, P.: Explicit fusions. TCS 340(3), 606–630 (2005)
19. Buscemi, M., Montanari, U.: Cc-pi: A constraint-based language for specifying service level agreements. In: De Nicola, R. (ed.) ESOP 2007. LNCS, vol. 4421, pp. 18–32. Springer, Heidelberg (2007)
20. Baldan, P., Corradini, A., Ehrig, H., Heckel, R., König, B.: Bisimilarity and behaviour-preserving reconfiguration of open petri nets. In: Mossakowski, T., Montanari, U., Haveraaen, M. (eds.) CALCO 2007. LNCS, vol. 4624, pp. 126–142. Springer, Heidelberg (2007)

21. Corradini, A., Große-Rhode, M., Heckel, R.: A coalgebraic presentation of structured transition systems. TCS 260, 27–55 (2001)
22. Corradini, A., Heckel, R., Montanari, U.: Tile transition systems as structured coalgebras. In: Proc. of FCT, pp. 13–38 (1999)
23. Bonchi, F.: Abstract Semantics by Observable Contexts. PhD thesis (2008)
24. Bonchi, F., Montanari, U.: Coalgebraic models for reactive systems. In: Caires, L., Vasconcelos, V.T. (eds.) CONCUR 2007. LNCS, vol. 4703, pp. 364–379. Springer, Heidelberg (2007)
25. Leifer, J.J., Milner, R.: Deriving bisimulation congruences for reactive systems. In: Palamidessi, C. (ed.) CONCUR 2000. LNCS, vol. 1877, pp. 243–258. Springer, Heidelberg (2000)
26. Hausmann, D., Mossakowski, T., Schröder, L.: A coalgebraic approach to the semantics of the ambient calculus. TCS 366(1-2), 121–143 (2006)
27. Bonchi, F., Gadducci, F., Monreale, G.V.: Reactive systems, barbed semantics and the mobile ambients. In: de Alfaro, L. (ed.) FOSSACS 2009. LNCS, vol. 5504, pp. 272–287. Springer, Heidelberg (2009)
28. Montanari, U., Pistore, M.: Finite state verification for the asynchronous pi-calculus. In: Cleaveland, W.R. (ed.) TACAS 1999. LNCS, vol. 1579, pp. 255–269. Springer, Heidelberg (1999)
29. Pistore, M., Sangiorgi, D.: A partition refinement algorithm for the pi-calculus. Information and Computation 164(2), 264–321 (2001)