A congruence result for process calculi with structural axioms

Maria Grazia Buscemi and Ugo Montanari

Dipartimento di Informatica, University of Pisa, Italy.
{buscemi,ugo}@di.unipi.it

Abstract. Bialgebraic models of process calculi enjoy the property that bisimilarity is a congruence. Indeed, the unique morphism to the final bialgebra induces a bisimilarity relation which coincides with observational equivalence and which is a congruence with respect to the operations. However, the application of the bialgebraic approach to process calculi with structural axioms is more problematic, because of the interaction between axioms and inference rules. In this paper, we generalise a previous method proposed by the same authors to lift calculi with structural axioms to bialgebras. In order for the lifting to hold, two conditions are required: the transition rules of the calculus are in TYFT format and the axioms bisimulate with respect to the lts. As a simple example of applicability of this general approach we consider CCS with replication, thus providing a compositional bialgebraic model of the calculus.

1 Introduction

Structural operational semantics [1, 16] is a well-established technique to provide process calculi and specification languages with an interpretation. A transition system is inductively derived from a set of transition rules that describe the behaviour of every construct of the language.

Transition systems can be easily seen as coalgebras for a functor in the category **Set**. A coalgebraic framework presents several advantages; for instance, morphisms between coalgebras (cohomomorphisms) enjoy the property of "reflecting behaviours" and thus they allow to characterise bisimulation equivalences as kernels of morphisms and bisimilarity as the kernel of the morphism to the final coalgebra [17].

In the above representation of transition systems, states are seen as just forming a set and the algebraic structure modelling the construction of programs and the composition of states is disregarded. On the other side, on transition systems with an algebraic structure on states, equivalences which are congruences with respect to the operators are useful to provide a compositional abstract semantics.

The missing algebraic structure may be recovered by integrating coalgebras with algebras, as it is done by Turi and Plotkin [19]. They propose a *bialgebraic semantics* based on the classical approach to syntax and a coalgebraic approach

Research supported in part by FET Global project *PROFUNDIS*

to behaviours, and show how to derive, in an abstract fashion, GSOS as a congruence format for bisimulation equivalence. Roughly, bialgebras are structures that can be regarded both as algebras of coalgebras and as coalgebras of algebras. Morphisms between bialgebras are both algebra homomorphisms and coalgebra cohomomorphisms and, thus, the unique morphism to the final bialgebra, which exists under reasonable assumptions, induces a (coarsest) bisimulation congruence on any coalgebra.

The compositional nature of bialgebras also gives advantages in finite state verification methods [5]. Usually, such techniques glue components together and, then, apply model checking or other verification methods. However, in many cases, state explosion severely limits the applicability. A compositional approach, on the other side, gives a chance to minimize components before combining them, thus preventing state explosion, at some extent, and yielding a smaller state space.

In this paper we generalise a previous method proposed by the same authors [2] to lift calculi with structural axioms to bialgebras. We show that, in order for the lifting to hold, two conditions are required: the transition rules of the calculus must be in TYFT format [8] and the axioms must bisimulate with respect to the transition system. The construction in [2] is limited to rules in the more specialised De Simone format and is exploited to provide a coalgebraic model of the early pi-calculus. Actually, since early-bisimilarity is not a congruence, it is not surprising that this model is not fully compositional, in the sense that some operators, remarkably input, are treated as constants.

We first consider a category $\mathbf{Alg}(\Sigma)$ of Σ-algebras and Σ-morphisms and show that the powerset functor P_L can be lifted to a functor P_Δ on $\mathbf{Alg}(\Sigma)$ using the TYFT specification. Next, we prove that, given a Σ-algebra B with a complete axiomatization E, a coalgebra $g : |B| \to P_L(|B|)$ can be lifted to a bialgebra $g : B \to P_\Delta(B)$, provided that each axiom in E bisimulates. By the theory on bialgebras, it then follows that bisimilarity is a congruence.

As an example of applicability of this method we consider CCS [11] with replication. We define a transition system with SOS rules in TYFT format, and show that it can be lifted to a bialgebra. This is a simple example, since, for the purpose of this paper, we are only interested in showing how the method works. Of course, calculi with more complex axiomatizations can be modelled within our framework. For instance, the coalgebraic model of early pi-calculus presented in [2] is based on the lifting result. The main idea behind that model is to consider an algebra of permutations enriched with a name extrusion operator à la De Brujin, that shifts any name to the successor and generates a new name in the first variable x_0. The set of axioms of the associated algebraic specification, thus, not only contains standard axioms of the pi-calculus, but also axioms for dealing with permutations and name extrusion operator.

A lifting method has also been proposed in [4], though following a different line of reasoning. The construction in [4] yields a category of bialgebras which satisfy a set of axioms, that is, coalgebras on the category $\mathbf{Alg}(\Gamma)$ of algebras satisfying a specification $\Gamma = (\Sigma, E)$. In our approach, on the contrary, the algebraic specification is only the signature without the axioms, thus our treatment turns out to be simpler: the lifting of the functor, for instance, derives from the fact that the SOS rules are in TYFT format, and the essence of the proof is showing that g is a morphism in $\mathbf{Alg}(\Sigma)$. However, the final coalgebra we obtain is larger but, given a coalgebra which satisfies the axioms, its image in the final coalgebra is the same in both the approaches.

The results presented in [4] have been applied in [13, 14] to give a coalgebraic semantics of the π-calculus. This model is flat, in the sense that it represents the calculus at an operational level, and exploits permutation algebras: the intuition is that the effects of permutations on the behaviour of agents are the smallest information required to define observational equivalence via ordinary bisimilarity, without limitations due to name generation and passing.

Klin [9, 10] has proposed an alternative approach for modelling process equivalences, based on notions of test and test suites. This technique has been combined with the bialgebraic framework of Turi and Plotkin, yielding a method for deriving congruence format for different process equivalences and, in particular, completed trace equivalence.

This paper is organised as follows. Section 2 contains the basic theory on coalgebras and bialgebras. In Sect. 3 we prove a general result to lift coalgebras in **Set** to coagebras on $\mathbf{Alg}(\Sigma)$. In Sect. 4 we consider CCS with replication as a small example of application of our lifting method. Sect. 5 contains some concluding remarks.

2 Coalgebraic/bialgebraic semantics

In this section we present the relevant definitions and results about coalgebras and bialgebras. We start reviewing notions about algebras and algebraic specifications.

We recall that, for Σ a signature, a Σ-algebra $A = \langle |A|, (op^A)_{op \in \Sigma} \rangle$ consists of a carrier set $|A|$ and a family of operations such that $op^A : |A|^n \to |A|$ if $op \in \Sigma$ of arity n. We assume to have a countable set X of the variables that can be used in the terms of the algebra. A Σ-homomorphism (or simply a morphism) between two Σ-algebras A and B is a function $h : |A| \to |B|$ that commutes with all the operations in Σ, namely, for each operator $op \in \Sigma$ of arity n, we have $op^B(h(a_1), \cdots, h(a_n)) = h(op^A(a_1, \ldots, a_n))$. We denote by $\mathbf{Alg}(\Sigma)$ the category of *Σ-algebras* and *Σ-morphisms*. A Σ-algebra A satisfies an algebraic specification $\Gamma = \langle \Sigma, E \rangle$, if A satisfies all axioms in E. In this case, A is called a Γ-algebra. The category of Γ-algebras and homomorphisms is the full subcategory $\mathbf{Alg}(\Gamma) \subseteq \mathbf{Alg}(\Sigma)$.

The basic idea behind SOS specifications is to specify a transition relation by induction over the structure of the system's states. In order to make explicit this structure, rather than ordinary labelled transition systems we consider transition systems whose sets of states have an algebraic structure.

Definition 1 (transition systems). *Let $\Gamma = \langle \Sigma, E \rangle$ be an algebraic specification, and L be a set of labels. A* labelled transition system (transition system, *in* *brief*) *over* Γ *and* L *is a pair* $lts = \langle A, \longrightarrow_{lts} \rangle$ *where* A *is a nonempty* Γ*-algebra and* $\longrightarrow_{lts} \subseteq |A| \times L \times |A|$ *is a labelled transition relation. For* $\langle p, l, q \rangle \in \longrightarrow_{lts}$ *we write* $p \xrightarrow{l}_{lts} q$.

Let $lts = \langle A, \longrightarrow_{lts} \rangle$ *and* $lts' = \langle B, \longrightarrow_{lts'} \rangle$ *be two transition systems. A morphism* $h : lts \to lts'$ *of transition systems over* Γ *and* L *(lts morphism, in brief) is a* Γ*-morphism* $h : A \to B$ *such that* $p \xrightarrow{l}_{lts} q$ *implies* $f(p) \xrightarrow{l}_{lts'} f(q)$.

The notion of bisimulation and transition systems with an algebraic structure is the classical one.

Definition 2 (bisimulation). *Let* $\Gamma = \langle \Sigma, E \rangle$ *be an algebraic specification,* L *be a set of labels, and* $lts = \langle A, \longrightarrow_{lts} \rangle$ *be a transition system over* Γ *and* L.

A relation $\mathcal{R}$ *over* $|A|$ *is a* bisimulation *if* $p \mathrel{\mathcal{R}} q$ *implies:*

- *for each* $p \xrightarrow{l} p'$ *there is some* $q \xrightarrow{l} q'$ *such that* $p' \mathrel{\mathcal{R}} q'$*;*
- *for each* $q \xrightarrow{l} q'$ *there is some* $p \xrightarrow{l} p'$ *such that* $p' \mathrel{\mathcal{R}} q'$*.*

Bisimilarity $\sim_{lts}$ *is the largest bisimulation.*

Given an algebraic specification $\Gamma = \langle \Sigma, E \rangle$ and a set of labels L, a collection of SOS rules can be regarded as a specification of those transition systems over Γ and L that have a transition relation closed under the given rules.

Definition 3 (SOS rules). *Given an algebraic specification* $\Gamma = \langle \Sigma, E \rangle$ *and a set of labels* L*, a sequent* $p \xrightarrow{l} q$ *(over* L *and* Γ*) is a triple where* $l \in L$ *is a label and* p, q *are* Σ*-terms with variables in a given set* X*.*

An SOS rule r *over* Γ *and* L *takes the form:*

$$\frac{p_1 \xrightarrow{l_1} q_1 \quad \cdots \quad p_n \xrightarrow{l_n} q_n}{p \xrightarrow{l} q}$$

where $p_i \xrightarrow{l_i} q_i$ *for all* $i = 1, \cdots n$ *as well as* $p \xrightarrow{l} q$ *are sequents.*

We say that transition system $lts = \langle A, \longrightarrow_{lts} \rangle$ satisfies *a rule* r *like above if each assignment to the variables in* X *that is a solution*[1] *to* $p_i \xrightarrow{l_i} q_i$ *for* $i = 1, \ldots, n$ *is also a solution to* $p \xrightarrow{l} q$.

Definition 4 (transition specifications). *A* transition specification *is a tuple* $\Delta = \langle \Gamma, L, R \rangle$ *consisting of an algebraic specification* Γ*, a set of labels* L*, and a set of SOS rules* R *over* Γ *and* L*. We abbreviate* $\Delta = \langle \Gamma = \langle \Sigma, \emptyset \rangle, L, R \rangle$ *by* $\Delta = \langle \Sigma, L, R \rangle$

A transition system over Δ *is a transition system over* Γ *and* L *that satisfies rules* R*.*

---

[1] Given $h : X \to A$ and its extension $\hat{h} : T_{(\Sigma,E)}(X) \to A$, h is a solution to $p \xrightarrow{l} q$ for *lts* if and only if $\hat{h}(p) \xrightarrow{l}_{lts} \hat{h}(q)$.

It is well known that ordinary labelled transition systems (i.e., transition systems whose states do not have an algebraic structure) can be represented as coalgebras for a suitable functor [17].

Definition 5 (coalgebras). *Let $F : \mathcal{C} \to \mathcal{C}$ be a functor on a category $\mathcal{C}$. A* coalgebra *for F, or F-*coalgebra*, is a pair $\langle A, f \rangle$ where A is an object and $f : A \to F(A)$ is an arrow of $\mathcal{C}$. A F-*cohomomorphism *(or simply F-*morphism*) $h : \langle A, f \rangle \to \langle B, g \rangle$ is an arrow $h : A \to B$ of $\mathcal{C}$ such that*

$$h; g = f; F(h). \tag{1}$$

We denote with $\mathbf{Coalg}(F)$ the category of F-coalgebras and F-morphisms.

Proposition 1. *For a fixed set of labels L, let $P_L : \mathbf{Set} \to \mathbf{Set}$ be the functor defined on objects as $P_L(X) = \mathcal{P}(L \times X + X)$, where $\mathcal{P}$ denotes the countable powerset functor, and on arrows as $P_L(h)(S) = \{\langle l, h(p)\rangle \mid \langle l, p\rangle \in S \cap L \times X\} \cup \{h(p) \mid p \in S \cap X\}$, for $h : X \to Y$ and $S \subseteq L \times X + X$. Then P_L-coalgebras are in a one-to-one correspondence with transition systems[2] on L, given by $f_{lts}(p) = \{\langle l, q\rangle \mid p \xrightarrow{l}_{lts} q\} \cup \{p\}$ and, conversely, by $p \xrightarrow{l}_{lts_f} q$ if and only if $\langle l, q\rangle \in f(p)$.*

In [3] the generalised notion of lax cohomomorphism is given, in order to accommodate also the more general definition of lts morphisms in a (lax) coalgebraic framework. To make clear their intuition, let $f : A \to P_L(A)$ and $g : B \to P_L(B)$ be two P_L-coalgebras and let $h : A \to B$ be a P_L-morphism. If we split the morphism condition (1) for h in the conjunction of the two inclusions $f; P_L(h) \subseteq h; g$ and $h; g \subseteq f; P_L(h)$, then it is easily shown that the first inclusion expresses "preservation" of transitions, while the second one corresponds to "reflection". Thus, lts morphisms can be seen as arrows (i.e., functions in **Set**) that satisfy the first inclusion, while lts morphisms which also satisfy the reflection inclusion are P_L-morphisms. This observation will be useful in Sect. 3.

Definition 6 (TYFT format). *Given an algebraic specification $\Gamma = \langle \Sigma, E\rangle$ and a set of labels L, a rule r over Γ and L is in* TYFT format *if it has the form:*

$$r = \frac{\{t_i \xrightarrow{l_i} y_i\}_{i \in I}}{op(x_1, \ldots, x_n) \xrightarrow{l} t}$$

where $op \in \Sigma$, y_i and x_j are all distinct variables.

Rule r is pure *if all its variables are among $\{y_i\}_{i\in I} \cup \{x_j\}_{1\le j \le n}$. Furthermore, r is* look-ahead free *if for all $i \in I$, $Var(p_i) \subseteq \{x_1, \ldots, x_n\}$. In the rest of the paper, we will only consider* TYFT *rules which are pure and look-ahead free and, for brevity, we will simply call them '*TYFT *rules'.*

A TYFT proof *of sequent $s \xrightarrow{l} t$ from premises $\{x_i \xrightarrow{l_i} y_i\}_{i\in I}$ is a proof of $s \xrightarrow{l} t$ from $\{x_i \xrightarrow{l_i} y_i\}_{i \in I}$ that is obtained using only* TYFT *rules in R and without using axioms in E.*

---

[2] Notice that this correspondence is well defined also for transition systems with sets of states, rather than with algebras of states as required in Definition 1.

The following results are due to Turi and Plotkin ([19]) and concern *bialgebras*, that is, coalgebras in $\mathbf{Alg}(\Sigma)$. As noted in the introduction, bialgebras enjoy the property that the unique morphism to the final bialgebra, which exists under reasonable conditions, induces a bisimulation that is a congruence with respect to the operations. The theory developed in [19] concerns transition systems in GSOS format. However, for the purpose of the paper, it will be enough to recall a simplified theory where rules are in TYFT format (remark that we abuse notation by writing 'TYFT rules' to mean pure, look-ahead free TYFT rules).

Proposition 2 (lifting of P_L). *Let $\Delta = \langle \Sigma, L, R\rangle$ be a transition specification with* TYFT *rules. Define $P_\Delta : \mathbf{Alg}(\Sigma) \to \mathbf{Alg}(\Sigma)$ as follows:*

- $|P_\Delta(A)| = P_L(|A|)$;
- *whenever* $\dfrac{\{x_i \xrightarrow{l_i} y_i \mid i \in I\}}{op(x_1, \ldots, x_n) \xrightarrow{l} p} \in R$ *then*
$$\frac{\langle l_i, p_i\rangle \in S_i,\ i \in I \quad q_j \in S_j,\ j \notin I}{\langle l, p[p_i/y_i,\ i \in I,\ \ q_j/y_j,\ j \notin I]\rangle \in op^{P_\Delta(A)}(S_1, \ldots, S_n)};$$
- *if $h : A \to B$ is a morphism in $\mathbf{Alg}(\Sigma)$ then $P_\Delta(h) : P_\Delta(A) \to P_\Delta(B)$ and* $P_\Delta(h)(S) = \{\, \langle l, h(p)\rangle \mid \langle l, p\rangle \in S \cap (L \times |A|)\,\} \cup \{\, h(p) \mid p \in S \cap |A|\,\}$.

Then P_Δ is a well-defined functor on $\mathbf{Alg}(\Sigma)$.

Corollary 1. *Let $\Delta = \langle \Sigma, L, R\rangle$ be a transition specification with rules R in* TYFT *format.*

Any morphism $h : f \to g$ in $\mathbf{Coalg}(P_\Delta)$ entails a bisimulation $\sim_h$ on lts_f, that coincides with the kernel of the morphism. Bisimulation $\sim_h$ is a congruence for the operations of the algebra.

Moreover, the category $\mathbf{Coalg}(P_\Delta)$ has a final object. Finally, the kernel of the unique P_Δ-morphism from f to the final object of $\mathbf{Coalg}(P_\Delta)$ is a relation on the states of f which coincides with bisimilarity on lts_f and is a congruence.

Note that the above corollary assumes that a P_L-coalgebra can be lifted to a P_Δ-coalgebra. Indeed, such assumption is obvious in the particular case of $f : A \to P_\Delta(A)$, with $A = T_\Sigma$ and f unique by initiality, namely when A has no structural axioms and no additional constants, and lts_f is the minimal transition system satisfying Δ. In the next section, we will spell out sufficient conditions to lift a transition system with structural axioms from $\mathbf{Coalg}(P_L)$ to $\mathbf{Coalg}(P_\Delta)$, under appropriate conditions on the axioms. Let us now consider an example that in general this lifting may not succeed.

Example 1. Consider a chemical abstract machine *CHAM*. For our purposes, the signature Σ_c of *CHAM* is defined as:

$$\Sigma_c \ ::= \ \mathbf{0} \ \big|\ \_|\_ \ \big|\ a.\_ \ \big|\ \bar{a}.\_ \ \big|\ \mathtt{redex}_a(\_,\_)$$

and E_c is the set of axioms for commutativity, associativity, *id*, $\mathbf{0}$ plus an axiom $\mathtt{redex}_a(p, q) = a.p \mid \bar{a}.q$. The only reduction rule $\mathtt{redex}_a(p, q) \longrightarrow p \mid q$ is TYFT.

For P_{Δ_c} the usual poweralgebra functor on $\mathbf{Alg}(\Sigma_c)$, the transition system of *CHAM* forms a P_L-coalgebra, but not a P_{Δ_c}-coalgebra [3]. And bisimilarity is not a congruence as, for example, $\mathbf{0} \sim a.p$ but $\bar{a}.p \not\sim a.p \,|\, \bar{a}.p$.

3 Lifting Coalgebras on Set to Bialgebras

In this section we prove a general result about lifting a coalgebra g in **Set** with set of states B and equipped with operations Σ, structural axioms E and TYFT rules R to a coalgebra in $\mathbf{Alg}(\Sigma)$. The lifting allows to apply Corollary 1 and, in particular, to prove that bisimilarity is a congruence.

Theorem 1. *Let $\mathcal{B}$ be the class of coalgebras g in* **Set** *with the following properties:*

1. $g : |B| \to P_L(|B|)$*, with* $B = T_{(\Sigma,E)}$*.*
2. lts_g *satisfies transition specification* $\Delta = (\langle \Sigma, E\rangle, L, R)$*, with R in* TYFT *format.*

Then, there is an initial coalgebra $\hat{g}$ in $\mathcal{B}$, such that $\forall g \in \mathcal{B}$, $\forall p \in B$, $p \xrightarrow{l}_{\hat{g}} q$ implies $p \xrightarrow{l}_{g} q$.

Furthermore, the transitions of $\hat{g}$ can be derived using the rules R and the following additional rule:

$$\text{(STRUCT)}\ \frac{t_1 =_E t_1' \quad t_1' \xrightarrow{l}_{\hat{g}} t_2' \quad t_2' =_E t_2}{t_1 \xrightarrow{l}_{\hat{g}} t_2}$$

where terms t_1, t_1', t_2, t_2' are in T_Σ.

Proof. Signature Σ, axioms E and transition specification Δ can be considered as a single algebraic specification, equipped with an initial model. The elements of the model can be derived using the proof system associated to the specification.

Definition 7. *Let $g : |B| \to P_L(|B|)$ be the initial coalgebra of Theorem 1. Then, we define the following Σ-algebras and Σ-morphisms:*

- $A = T_\Sigma$ *and $h : A \to B$ is the unique morphism in $\mathbf{Alg}(\Sigma)$ from the initial object;*
- $f : A \to P_\Delta(A)$ *is the unique arrow in $\mathbf{Alg}(\Sigma)$ from the initial object.*

In the sequel, we want to find conditions under which P_L-coalgebra g can be lifted to a P_Δ-coalgebra and function h, as above defined, to a P_Δ-morphism. The observation in Sect. 2 allows us to state, without any further condition, that h is a lax morphism between P_L-coalgebras f and g. The reflection inclusion, instead, will require appropriate hypotheses.

[3] Indeed, axiom $\texttt{redex}_a(p,q) = a.p \,|\, \bar{a}.q$ does not satisfy the "bisimulation" condition required in Theorem 2.

Property 1. Function h in Definition 7 is a lts morphism, namely, a lax P_L-coalgebra morphism. Furthermore, it is surjective.

Proof. Immediate, as every proof of a transition in lts_f holds also in lts_g.

Theorem 2. *Let g be the initial coalgebra in $\mathcal{B}$ as specified by Theorem 1, and let A, h, and f be defined as in Definition 7. Then, h is surjective. Let us assume that for all equations $t_1 = t_2$ in E, with free variables $\{x_i\}_{i \in I}$, we have* TYFT *proofs as follows (for t_1, t_1', t_2, t_2' terms of T_Σ):*

$$\frac{x_i \xrightarrow{l_i} y_i \quad i \in I}{t_1 \xrightarrow{l} t_1'} \quad \textit{implies} \quad \frac{x_i \xrightarrow{l_i} y_i \quad i \in I}{t_2 \xrightarrow{l} t_2'} \quad \textit{and} \quad t_1' =_E t_2' \tag{2}$$

and viceversa, using the rules in R.

Then, the left diagram below commutes in **Set***, i.e., $h; g = f; P_L(h)$. Thus, h is a P_L-morphism.*

Proof. We start noticing that $h : A \to B$ is surjective since $B = A_{/=_E}$ and $=_E$ is the kernel of h.

Then, we first prove that the equivalence relation $=_E$ is a bisimulation for lts_f. We use rule induction on the proofs of $=_E$. For axioms in E , the property is guaranteed by Condition 2. Rules for reflexivity, symmetry and transitivity are obviously satisfied. Rule for congruence is also easily checked, since if t_i and t_i' bisimulate, for $i = 1, \cdots, n$, also $op(t_1, \cdots, t_n)$ and $op(t_1', \cdots, t_n')$, with $k \in \Sigma_n$, have corresponding transitions by applying the same TYFT rule.

Since h is a lax P_L-coalgebra morphism, to derive our result it is enough to show that $h(t_1) \xrightarrow{l}_g p$ implies $t_1 \xrightarrow{l}_f t_2$, with $h(t_2) = p$. We prove this property by induction on the rules of lts_g, as specified by Theorem 1. Rules in R can be easily checked since they are the same for lts_f and for lts_g. Also, rule (STRUCT) preserves the property. Indeed, given $[t_1']_E \xrightarrow{l}_g [t_2']_E$, i.e., $h(t_1') \xrightarrow{l}_g h(t_2')$, by induction hypothesis we have $t_1' \xrightarrow{l}_f t_2''$, with $h(t_2'') = h(t_2')$. Furthermore, since $t_1 =_E t_1'$ and $=_E$ is a bisimulation for f, we can find $t_1 \xrightarrow{l}_f t_2'''$, with $t_2''' =_E t_2''$ and, thus, $h(t_2''') = h(t_2'')$. Also, $t_2' =_E t_2$ implies $h(t_2) = h(t_2')$. Then, $h(t_2''') = h(t_2)$.

$$\begin{array}{ccc} |A| & \xrightarrow{h} & |B| \\ {\scriptstyle f}\downarrow & & \downarrow{\scriptstyle g} \\ P_L(|A|) & \xrightarrow[P_L(h)]{} & P_L(|B|) \end{array} \qquad \begin{array}{ccc} A & \xrightarrow{h} & B \\ {\scriptstyle f}\downarrow & & \downarrow{\scriptstyle g} \\ P_\Delta(A) & \xrightarrow[P_\Delta(h)]{} & P_\Delta(B) \end{array}$$

Theorem 3. *Let g be the initial coalgebra in $\mathcal{B}$ as specified by Theorem 1, and let A, h, and f be defined as in Definition 7. If the left diagram above commutes in* **Set***, i.e., $h; g = f; P_L(h)$, then g can be lifted from* **Set** *to* $\mathbf{Alg}(\Sigma)$ *and the right diagram commutes in* $\mathbf{Alg}(\Sigma)$.

Proof. See the Appendix.

Corollary 2. *Let g be the initial coalgebra in $\mathcal{B}$ as specified by Theorem 1, and let the right diagram above commute. Then in g bisimilarity is a congruence.*

Proof. The claim follows by Theorem 3 and Corollary 1.

4 CCS with replication

In this section, we apply the results of Sect. 3 to the CCS with replication operator, denoted $\mathrm{CCS}_!$. We will show that the transition system of $\mathrm{CCS}_!$ satisfies the 'bisimilarity' condition required by Theorem 2. As a consequence, bisimilarity is a congruence with respect to the operations.

Let $\mathfrak{N}$ be the countable set of names, ranged over by $x, y, \cdots$. A signature $\Sigma_!$ for the calculus is defined as follows:

$$\Sigma_! ::= \mathbf{0} \mid \alpha.\_ \mid \_+\_ \mid \_\,|\,\_ \mid (\nu x)\,\_ \mid !\,\_ \qquad \alpha ::= \tau \mid x \mid \bar{x}$$

For input and output actions, we write $\bar{\alpha}$ for the complementary of α; that is, if $\alpha = x$ then $\bar{\alpha} = \bar{x}$, if $\alpha = \bar{x}$ then $\bar{\alpha} = x$. The occurrence of x in $(\nu x)\,p$ is bound; *free names* and *bound names* of agent term p are defined as usual and we denote them with $\mathrm{fn}(p)$ and $\mathrm{bn}(p)$, respectively. Also, we denote with $\mathrm{n}(p)$ the sets of (free and bound) names of agent term p. We take the calculus equipped with a set of axioms $E_!$, consisting of axioms for commutativity, associativity, *id*, $\mathbf{0}$ plus an axiom for replication:

$$p \;|!\,p \equiv !\,p.$$

Definition 8 (transition specification for $\mathrm{CCS}_!$). *We define a transition specification $\Delta_!$ for $\mathrm{CCS}_!$ as $\Delta_! = \langle \Sigma_!, L_!, R_! \rangle$, where $\Sigma_!$ is the signature defined above, $L_!$ is a set of* labels $L_! = \{\tau,\ x,\ \overline{x}\ ,\ \mid x, y \text{ names}\}$*; $R_!$ is the set of SOS rules presented in Table 1.*

(Pre)	$\alpha.p \xrightarrow{\alpha} p$	(Sum)	$\dfrac{p \xrightarrow{\alpha} p'}{p+q \xrightarrow{\alpha} p'}$
(Par)	$\dfrac{p \xrightarrow{\alpha} p'}{p\|q \xrightarrow{\alpha} p'\|q}$	(Com)	$\dfrac{p \xrightarrow{\bar{x}} p' \quad q \xrightarrow{x} q'}{p\|q \xrightarrow{\tau} p'\|q'}$
(Rep)	$\dfrac{p\|!p \xrightarrow{\alpha} p'}{!p \xrightarrow{\alpha} p'}$	(Res)	$\dfrac{p \xrightarrow{\alpha} q}{(\nu x)\,p \xrightarrow{\alpha} (\nu x)\,q}$ if $x \notin \mathrm{n}(\alpha)$

Table 1. SOS rules for $\mathrm{CCS}_!$ (symmetric rules are omitted)

Proposition 3. *The rules of the transition specification of* $\mathrm{CCS}_!$ *that are defined in Table 1 are in* TYFT *format.*

Note that rule (Rep) is not De Simone format, since the source of the premise is not a variable, but a complex term.

Definition 9 (lts for $\mathrm{CCS}_!$). *A transition system $lts_{\mathrm{CCS}_!}$ for $\mathrm{CCS}_!$ is defined as $\langle T_{(\Sigma_!, E_!)}, \longrightarrow \rangle$, where signature $\Sigma_!$ and axioms $E_!$ are as defined above, and $\longrightarrow$ is defined by the SOS rules in Table 1.*

Theorem 4. *Bisimilarity* $\sim_{lts_{\mathrm{CCS}_!}}$ *in* CCS$_!$ *is a congruence.*

Proof. (Hint) By Theorem 2, we have to prove that axioms $E_!$ satisfy the 'bisimilarity' condition. The fact that axiom for replication bisimulates follows by rule (Rep), while it is well-known that the other axioms bisimulate (see [11], for instance).

Note that, for the sake of simplicity, we have not considered the CCS relabelling operator. This construct can, however, be accommodated within our framework, by enriching both the transition specification and the algebra for CCS$_!$ with permutations and proving that the extra axioms bisimulate with respect to the extended transition system (see [14, 2] for details).

5 Conclusions

We have proposed a method to provide compositional coalgebraic models of process calculi with structural axioms, under appropriate conditions. Specifically, we have proved that the transition system of a calculus can be lifted to a bialgebra, provided that the SOS rules are in tyft format and the axioms bisimulate.

We plan to apply this method to open pi-calculus [18]. The main difficulty arises in modelling *distinctions*, that is relations that specify which names cannot be fused. Our proposal is to generalise the permutation algebra considered for the pi-calculus [2] with (possibly, non-injective) substitutions and to introduce *types* to model distinctions. Moreover, since open bisimilarity is a congruence, we expect this substitution algebra can be enriched with the full set of operations of the calculus (remarkably, including input prefix).

References

1. L. Aceto, W.J. Fokkink, and C. Verhoef. Structural Operational Semantics. In *Handbook of Process Algebra*, Elsevier, 2002.
2. M. Buscemi and U. Montanari. A First Order Coalgebraic Model of Pi-Calculus Early Observational Equivalence. In *Proc. of CONCUR '02*, LNCS 2421, Springer, 2002.
3. A. Corradini, M. Große-Rhode, R. Heckel. Structured transition systems as lax coalgebras. In *Proc. of CMCS'98*, ENTS 11. Elsevier Science, 1998.
4. A. Corradini, R. Heckel, and U. Montanari. Compositional SOS and beyond: A coalgebraic view of open systems. *Theoretical Computer Science* 280:163-192, 2002.
5. G. Ferrari, U. Montanari, and M. Pistore. Minimizing Transition Systems for Name Passing Calculi: A Co-algebraic Formulation. In *Proc. of FoSSaCS'02*, LNCS 2303. Springer, 2002.
6. M. Fiore and D. Turi. Semantics of name and value passing. In *Proc. of LICS'01*, IEEE. Computer Society Press, 2001.
7. M. Gabbay and A. Pitts. A new approach to abstract syntax involving binders. In *Proc. of LICS'99*, IEEE. Computer Society Press, 1999.
8. J. F. Groote and F. W. Vaandrager. Structured operational semantics and bisimulation as a congruence. *Information and Computation*, 100(2):202–260, 1992.

9. B. Klin. An Abstract Coalgebraic Approach to Process Equivalence for Well-Behaved Operational Semantics. PhD dissertation, Aarhus University, 2004.
10. B. Klin. From Bialgebraic Semantics to Congruence Formats. In SOS'04, ENTCS (to appear).
11. R. Milner. *Communication and Concurrency*. Prentice-Hall, 1989.
12. R. Milner, J. Parrow, and D. Walker. A calculus of mobile processes (parts I and II). *Information and Computation*, 100(1):1–77, 1992.
13. U. Montanari and M. Pistore. Pi-Calculus, Structured Coalgebras and Minimal HD-Automata. In *Proc. of MFCS'00*, LNCS 1983. Springer, 2000.
14. U. Montanari and M. Pistore. Structured Coalgebras and Minimal HD-Automata for the pi-Calculus. Technical Report 0006-02, IRST-ITC, 2000. Available at the URL: `http://sra.itc.it/paper.epl?id=MP00`.
15. A. M. Pitts. Nominal Logic: A First Order Theory of Names and Binding. In *Proc. of TACS'01*, LNCS 2215. Springer, 2001.
16. G. D. Plotkin. A Structural Approach to Operational Semantics. *Journal of Logic and Algebraic Programming*, 60-61: 17–139, 2004.
17. J. J. M. M. Rutten. Universal coalgebra: a theory of systems. *Theoretical Computer Science* 249(1):3–80, 2000.
18. D. Sangiorgi. A Theory of Bisimulation for the pi-Calculus. *Acta Informatica*, 33(1): 69–97, 1996.
19. D. Turi and G. Plotkin. Towards a mathematical operational semantics. In *Proc. of LICS'97*, IEEE. Computer Society Press, 1997.