Available online at www.sciencedirect.com
SCIENCE DIRECT®
Electronic Notes in Theoretical Computer Science
ELSEVIER Electronic Notes in Theoretical Computer Science 106 (2004) 201–218
www.elsevier.com/locate/entcs

A Coalgebraic Approach to Process Equivalence and a Coinduction Principle for Traces

Bartek Klin[1]

BRICS, Aarhus University[2]

Abstract

An abstract coalgebraic approach to well-structured relations on processes is presented, based on notions of tests and test suites. Preorders and equivalences on processes are modelled as coalgebras for behaviour endofunctors lifted to a category of test suites. The general framework is specialized to the case of finitely branching labelled transition systems. It turns out that most equivalences from the so-called van Glabbeek spectrum can be described by well-structured test suites. As an immediate application, coinductive proof principles are described for these equivalences, in particular for trace equivalence.

Keywords: van Glabbeek spectrum, coalgebra, coinduction.

1 Introduction

In the theory of concurrent processes as transition systems, various operational equivalences and preorders on processes are considered, corresponding to different notions of "similarity" between processes. In the standard case of nondeterministic, labelled transition systems, the most popular notions include bisimulation equivalence, simulation preorder, trace equivalence, failures equivalence etc. These notions have been studied thoroughly in [3] and are collectively known as the *van Glabbeek spectrum*. For other kinds of systems,

[1] Email: mailto:klin@brics.dk

[2] Basic Research in Computer Science (http://www.brics.dk), funded by the Danish National Research Foundation.

doi:10.1016/j.entcs.2004.02.029

known equivalences include various kinds of weak bisimulation equivalences, probabilistic bisimulation equivalence and many others.

In the coalgebraic approach to the theory of processes (cf. [15] as the main reference), the notion of transition system is parametrized by a notion of behaviour, modelled as an endofunctor B. Systems are then defined to be B-coalgebras. For example, if $BX = \mathcal{P}_{\mathrm{f}}(A \times X)$ on **Set** (for a fixed set A of actions), B-coalgebras correspond to finitely branching labelled transition systems.

The coalgebraic approach allows one to describe abstractly several kinds of transition systems. Given the importance of various operational equivalences in process algebra, an equally abstract treatment of these equivalences is desirable. Several attempts in this direction have been made (see Section 7), but even in the case of labelled transition systems, none have covered all equivalences and preorders from the van Glabbeek spectrum so far.

In [8,9], a novel abstract coalgebraic approach to process equivalence was presented, based on the notions of tests and test suites. There, equivalences and preorders are modelled as coalgebras for behaviour endofunctors suitably lifted to a category of test suites **TS**. Intuitively, two processes are considered equivalent if they cannot be distinguished by means of any test from a given test suite. Varying the test suites considered, different notions of process equivalence are obtained.

This general technique was applied to the case of finitely branching labelled transition systems, and a test suite characterization of three equivalences from the van Glabbeek spectrum (trace equivalence, completed trace equivalence and failures equivalence) was provided. Moreover, it was shown how to combine the test suite approach with bialgebraic methods of [17] to systematically derive congruence formats of structural operational semantics.

This paper makes two new contributions. Firstly, the test suite framework is showed to cover three other equivalences from the spectrum: ready trace equivalence, simulation equivalence and bisimulation equivalence. Moreover, the characterization of these equivalences as the least coalgebras for suitably lifted behaviour endofunctors is obtained in a more structured fashion than in [9], via a correspondence between tests and modal formulae in the respective fragments of the Hennessy-Milner logic. This hopefully will convince the reader that other notions from the van Glabbeek spectrum can be characterized in a similar manner.

Secondly, full relational characterizations of *all* (and not only least ones, as in [8]) coalgebras for lifted behaviour endofunctors are provided. This gives rise to novel, specialized coinduction proof principles for various operational equivalences.

For example, a notion of Tr-*aware relation* is defined, such that (1) trace preorder is the largest Tr-aware relation, and (2) Tr-aware relations have enough structure to facilitate proofs that given relations are Tr-aware. This allows one to prove coinductively that two given processes are trace equivalent, by showing any Tr-aware relations that relate them. Thus Tr-aware relations are connected to trace equivalence in the same way as bisimulations are connected to bisimulation equivalence, or simulations to simulation equivalence. The notion of Tr-aware relation is obtained by careful analysis of coalgebras for a suitably lifted endofunctor on $\mathbf{TS}$.

After Section 2 of preliminaries, the abstract framework of test suites is recalled in Section 3. In Section 4, it is specialized to the behaviour $BX = \mathcal{P}_{\mathrm{f}}(A \times X)$, and several equivalences from the van Glabbeek spectrum are related to coalgebras for B suitably lifted to the category $\mathbf{TS}$. In Section 5, these coalgebras are studied in more detail, which leads to the definition of various coinduction principles. A simple application of one of those principles (that corresponding to trace equivalence) is shown in Section 6. A brief description of related work is given in Section 7.

Acknowledgements. The author is very grateful to Gordon Plotkin for inspiration and encouragement, to Pawel Sobocinski for many discussions, and to anonymous referees for useful remarks.

2 Preliminaries

In this section, we present standard notions and results related to labelled transition systems, operational equivalences and coinduction, taken mostly from [3] and [15].

Definition 2.1 A *labelled transition system* (LTS) $\langle X, A, \longrightarrow\rangle$ is a set X of *processes*, a set A of *actions*, and a *transition relation* $\longrightarrow \subseteq X \times A \times X$.

As usual, we will write $x \xrightarrow{a} x'$ instead of $\langle x, a, x'\rangle \in \longrightarrow$. For any $x \in X$, one defines the set of *initials* $I(x) = \Big\{ a \in A \,:\, x \xrightarrow{a} x' \text{ for some } x' \in X \Big\}$.

An LTS $\langle X, A, \longrightarrow\rangle$ is *finitely branching*, if for every process $x \in X$ there are only finitely many processes $x' \in X$ and actions $a \in A$ such that $x \xrightarrow{a} x'$.

An LTS for which its underlying graph (obtained by ignoring all actions) is a rooted, directed tree is called a *labelled synchronization tree*.

Definition 2.2 Given a set of actions A, one considers sets of *modal formulae* $\mathcal{F}_{\mathsf{Tr}}$, $\mathcal{F}_{\mathsf{RdTr}}$, $\mathcal{F}_{\mathsf{S}}$ and $\mathcal{F}_{\mathsf{BS}}$, given by the following BNF grammars (here a ranges over A, and Q ranges over subsets of A):

$$\mathcal{F}_{\mathsf{Tr}}\ \phi ::= \top \mid \langle a \rangle\, \phi \qquad \mathcal{F}_{\mathsf{RdTr}}\ \phi ::= \top \mid \langle a \rangle\, \phi \mid \check{Q} \wedge \langle a \rangle\, \phi$$

$$\mathcal{F}_{\mathsf{S}}\ \ \phi ::= \top \mid \langle a \rangle\, \phi \mid \phi \wedge \phi \qquad \mathcal{F}_{\mathsf{BS}}\ \ \phi ::= \top \mid \bot \mid \langle a \rangle\, \phi \mid [a]\phi \mid \phi \wedge \phi \mid \phi \vee \phi$$

Definition 2.3 Given an LTS $h = \langle X, A, \longrightarrow \rangle$, the satisfaction relation $\models_h$ between processes and modal formulae is defined inductively as follows:

$$\begin{array}{lcl}
x \models_h \top & \text{always} & x \models_h \bot \quad \text{never} \\
x \models_h \langle a \rangle\, \phi & \iff & x' \models_h \phi \text{ for some } x' \text{ such that } x \xrightarrow{a} x' \\
x \models_h [a]\phi & \iff & x' \models_h \phi \text{ for all } x' \text{ such that } x \xrightarrow{a} x' \\
x \models_h \check{Q} & \iff & I(x) = Q \\
x \models_h \phi_1 \wedge (\vee) \phi_2 & \iff & x \models_h \phi_1 \text{ and (or) } x \models_h \phi_2
\end{array}$$

The set of formulae $\mathcal{F}_{\mathsf{BS}}$ together with the above interpretation is called the (finitary) *Hennessy-Milner logic*.

Definition 2.4 For any $\mathsf{W} \in \{\mathsf{Tr}, \mathsf{RdTr}, \mathsf{S}, \mathsf{BS}\}$ one considers the respective *operational preorder* $\sqsubseteq_{\mathsf{W}} \subseteq X \times X$ and an *operational equivalence* $\cong_{\mathsf{W}} \subseteq X \times X$, defined on a given LTS h as follows:

$$x \sqsubseteq_{\mathsf{W}} x' \iff (\forall \phi \in \mathcal{F}_{\mathsf{W}}.\ x \models_h \phi \Longrightarrow x' \models_h \phi)$$

$$x \cong_{\mathsf{W}} x' \iff (\forall \phi \in \mathcal{F}_{\mathsf{W}}.\ x \models_h \phi \iff x' \models_h \phi)$$

Preorders $\sqsubseteq_{\mathsf{Tr}}$ and $\sqsubseteq_{\mathsf{RdTr}}$ on a given LTS are usually called *trace preorder* and *ready trace preorder*, respectively. The corresponding equivalences are named in a similar manner. The preorders $\sqsubseteq_{\mathsf{S}}$, $\sqsubseteq_{\mathsf{BS}}$ and equivalences $\cong_{\mathsf{S}}$, $\cong_{\mathsf{BS}}$ are considered in more detail below.

In the following definitions, a given LTS $\langle X, A, \longrightarrow \rangle$ is assumed.

Definition 2.5 A relation $R \subseteq X \times X$ is a *simulation*, if xRy implies

- $\forall x \xrightarrow{a} x'.\ \exists y \xrightarrow{a} y'.\ x'Ry'$.

If, moreover, xRy implies

- $\forall y \xrightarrow{a} y'.\ \exists x \xrightarrow{a} x'.\ x'Ry'$.

then R is a *bisimulation*.

Definition 2.6 Processes $x, y \in X$ are

- in *simulation preorder*, if there exists a simulation R such that xRy,

- *simulation equivalent*, if there exist simulations R, R' such that xRy and $yR'x$,
- *bisimulation equivalent*, if there exists a bisimulation R such that xRy.

Proposition 2.7 Simulation preorder is indeed a preorder and it is the largest simulation on a given LTS. Bisimulation equivalence is indeed an equivalence relation and it is the largest bisimulation on a given LTS.

Proposition 2.8 In any LTS, the relation $\sqsubseteq_{\mathsf{S}}$ is equal to simulation preorder, the relation $\cong_{\mathsf{S}}$ is equal to simulation equivalence, and the relations $\sqsubseteq_{\mathsf{BS}}$ and $\cong_{\mathsf{BS}}$ are both equal to bisimulation equivalence.

The use of coalgebras in the abstract theory of processes has been motivated by an easy correspondence between $\mathcal{P}_{\mathrm{f}}(A \times -)$-coalgebras, (where A is a fixed set and $\mathcal{P}_{\mathrm{f}} : \mathbf{Set} \to \mathbf{Set}$ is the covariant finite powerset functor) and finitely branching labelled transition systems. Indeed, given an LTS $\langle X, A, \longrightarrow\rangle$, consider a function $h : X \to \mathcal{P}_{\mathrm{f}}(A \times X)$ defined by

$$\langle a, x'\rangle \in hx \iff x \xrightarrow{a} x'$$

It is easy to check that this gives a 1-1 correspondence. In the following we will often use this correspondence silently, identifying finitely branching LTSs with their corresponding coalgebras.

Varying the endofunctor B, one obtains similar correspondences between coalgebras and deterministic automata, labelled transition systems with state predicates, probabilistic transition systems and many others (see [15]).

The functor B used to represent transition systems as coalgebras is usually called a *behaviour endofunctor*. This notion only reflects the context of use of B, and does not restrict the class of functors considered.

It is well known that the finite powerset functor $\mathcal{P}_{\mathrm{f}}$ admits final coalgebras [2]. In particular:

Proposition 2.9 For any set A, the endofunctor $BX = \mathcal{P}_{\mathrm{f}}(A \times X)$ has a final coalgebra $\phi : \Omega \to B\Omega$ with Ω the set of (possibly infinite) finitely branching synchronization trees edge-labelled with elements of A, quotiented by bisimulation equivalence.

3 Test Suite Approach to Relations on Processes

In this section, we recall the test suite approach [8] to well-structured relations on processes, modelled as coalgebras. The approach has been inspired by unpublished ideas of Plotkin, who used topologies of tests to represent bisimulations on complete partial orders.

In Section 4, it will be shown how this approach allows to represent operational preorders and equivalences from the van Glabbeek spectrum.

Denote $2 = \{\mathtt{tt}, \mathtt{ff}\}$. A *test* on a set X is a function $V : X \to 2$. A *test suite* on a set X is a set of tests on X. The set of all test suites on X, partially ordered by (reverse) inclusion, is denoted X^*. For any function $f : X \to Y$, define the monotonic *reindexing function* $f^* : Y^* \to X^*$ by

$$f^*\theta = \{\, V \circ f \;:\; V \in \theta \,\}$$

where $\theta \in Y^*$. It is easy to check that this gives a functor $(-)^*$ from $\mathbf{Set}^{op}$ to the category $\mathbf{Pos}$ of partially ordered sets. On this functor one performs the well-known Grothendieck construction (see e.g. [5]), obtaining the following

Definition 3.1 The *test suite category* $\mathbf{TS}$ is defined as follows:

- objects in $\mathbf{TS}$ are pairs $\langle X, \theta \rangle$, where X is a set and $\theta \in X^*$,
- morphisms $f : \langle X, \theta \rangle \to \langle Y, \vartheta \rangle$ are functions $f : X \to Y$ such that $\theta \supseteq f^*\vartheta$.

Every test on X can be identified with a subset of X. However, we stick to the functional representation of tests, as it will make further developments (in particular, the definition of lifted endofunctor B^{W} below) look more natural. We will use T and F to denote the constantly true and the constantly false tests. We will also speak of unions and intersections of tests, denoted with $\vee$ and $\wedge$ and defined in the obvious way.

Any test suite induces two canonical specialization relations:

Definition 3.2 Let θ be a test suite on any set X. *Specialization equivalence* $\cong_\theta$ and *specialization preorder* $\leq_\theta$ are defined by

$$\cong_\theta = \{\langle x, y\rangle \in X \times X \mid \forall V \in \theta.\ Vx = Vy\}$$
$$\leq_\theta = \{\langle x, y\rangle \in X \times X \mid \forall V \in \theta.\ Vx = \mathtt{tt} \Rightarrow Vy = \mathtt{tt}\}$$

It is straightforward to show that test suite morphisms preserve specialization preorders and equivalences.

One says that an endofunctor $B^* : \mathbf{TS} \to \mathbf{TS}$ *lifts* an endofunctor $B : \mathbf{Set} \to \mathbf{Set}$ if $p \circ B^* = B \circ p$, where $p : \mathbf{TS} \to \mathbf{Set}$ is the obvious forgetful functor.

Any functor $B : \mathbf{Set} \to \mathbf{Set}$ can be lifted to an endofunctor on $\mathbf{TS}$ in possibly many ways, by defining its action on test suites. For our purposes, one particular way is especially useful. This well-structured method of lifting endofunctors is based on notions of test constructors and closures. Intuitively speaking, one might view tests as formulae interpreted on processes. Then test constructors correspond to modal operators in a language of formulae, and closures correspond to propositional connectives. More concrete examples supporting this intuition are presented in Section 4.

Definition 3.3 Let B be an endofunctor on **Set**. Tests on the set $B(2)$ are called *B-test constructors*.

Definition 3.4 A *test suite closure* is a family of monotonic functions $\mathrm{Cl}_X : X^* \to X^*$ for every set X, such that for any function $f : X \to Y$, and for any test suite θ on Y, one has $\mathrm{Cl}_X\, f^*\theta = f^*\, \mathrm{Cl}_Y\, \theta$.

Proposition 3.5 Let B be an endofunctor on **Set**. Any set of B-test constructors W[3], and any closure Cl induces a lifting of B to an endofunctor B^{W} on **TS**, defined by

$$B^{\mathsf{W}}\langle B, \theta\rangle = \left\langle BX, B_X^{\mathsf{W}}\theta\right\rangle \qquad B^{\mathsf{W}}f = Bf$$

where for any set X, the *action* $B_X^{\mathsf{W}} : X^* \to (BX)^*$ is a monotonic function defined by

$$B_X^{\mathsf{W}}\theta = \mathrm{Cl}_{BX}\{w \circ BV \mid w \in \mathsf{W}, V \in \theta\}$$

Let $B^{\mathsf{W}} : \mathbf{TS} \to \mathbf{TS}$ be defined as above. For any B-coalgebra $h : X \to BX$, define the operator $\Phi_h^{\mathsf{W}} : X^* \to X^*$ by

$$\Phi_h^{\mathsf{W}}\theta = h^* B_X \theta$$

It is clearly monotonic, since both h^* and B_X are monotonic.

B^{W}-coalgebras correspond to prefixed points of such operators: a B-coalgebra $h : X \to BX$ can be lifted to an B^{W}-coalgebra $h : \langle X, \theta\rangle \to \left\langle BX, B_X^{\mathsf{W}}\theta\right\rangle$ if and only if $\theta \supseteq \Phi_h^{\mathsf{W}}\theta$. In this case, one says that θ *lifts* h to a B^{W}-coalgebra.

Following the intuitive correspondence between tests and modal formulae, B^{W}-coalgebras are B-coalgebras equipped with sets of properties (predicates) closed under the modal operators from W and under the propositional connectives defined by Cl. The least such set corresponds to a certain modal logic interpreted on the underlying B-coalgebra. Again, this intuition will be confirmed in Section 4.

Proposition 3.6 If B has a final coalgebra $\phi : \Omega \to B\Omega$,then $\phi : \left\langle \Omega, \omega^{\mathsf{W}}\right\rangle \to \left\langle B\Omega, B_\Omega^{\mathsf{W}}\omega^{\mathsf{W}}\right\rangle$ is the final B^{W}-coalgebra, where ω^{W} is the least fixed point of Φ_ϕ^{W}.

Final B^{W}-coalgebras allow to construct, for any B-coalgebra $h : X \to BX$, the "least B^{W}-coalgebra that lifts h", i.e., the least test suite $\theta \in X^*$ that lifts h to an B^{W}-coalgebra:

[3] The relation of this W to the one used in Definition 2.4 will become clear in Section 4.2.

Proposition 3.7 Under the above notation, for any coalgebra $h : X \to BX$, the least fixed point of Φ_h^{W} (or, equivalently, the least element of X^* that lifts h to an B^{W}-coalgebra) is equal to $k^*\omega^{\mathsf{W}}$, where $k : X \to \Omega$ is the coinductive extension of h.

In the next section we will show how to represent various known equivalences and preorders on processes as specialization equivalences and preorders of the test suites $k^*\omega^{\mathsf{W}}$, for various liftings of behaviour to the category **TS**.

4 van Glabbeek Spectrum Described by Test Suites

In this chapter, we specialize the general framework described in Section 3 to describe various notions of operational equivalences and preorders from the van Glabbeek spectrum. For this purpose, fix the endofunctor $BX = \mathcal{P}_{\mathrm{f}}(A \times X)$ for a fixed set A.

In the following, various liftings B^{W} of B to **TS** will be proposed, based on different choices of sets W of B-test constructors. For different choices of W, B^{W}-coalgebras will be related, by means of specialization relations, to operational equivalences and preorders from the van Glabbeek spectrum. The correspondence will go via modal formulae in the respective fragments of the Hennessy-Milner logic, illustrating the correspondence between tests and formulae, test constructors and modal operators, and between closures and propositional connectives.

4.1 Test Constructors and Closures

We begin by defining some B-test constructors, useful to represent relations from the van Glabbeek spectrum.

Definition 4.1 For any $a \in A$, $Q \subseteq A$, define test constructors

$$w_{\langle a \rangle}, w_{[a]}, \check{w}_{aQ} : B2 \to 2$$

as follows:

$$\begin{aligned}
w_{\langle a \rangle}\beta = \mathtt{tt} &\iff \langle a, \mathtt{tt} \rangle \in \beta \\
w_{[a]}\beta = \mathtt{ff} &\iff \langle a, \mathtt{ff} \rangle \in \beta \\
\check{w}_{aQ}\beta = \mathtt{tt} &\iff w_{\langle a \rangle}\beta = \mathtt{tt} \text{ and } \{\, a \in A : \langle a, - \rangle \in \beta \,\} = Q
\end{aligned}$$

The following sets of test constructors will be useful:

Definition 4.2 Define

$$\begin{aligned}
\mathsf{Tr} &= \left\{\, w_{\langle a\rangle} \;:\; a \in A \,\right\} \\
\mathsf{RdTr} &= \mathsf{Tr} \cup \{\, \check{w}_{aQ} \;:\; a \in A, Q \subseteq A \,\} \\
\mathsf{BS} &= \mathsf{Tr} \cup \left\{\, w_{[a]} \;:\; a \in A \,\right\}
\end{aligned}$$

We will also consider three different closures:

Definition 4.3 Closures $\mathrm{Cl}^{\top}$, $\mathrm{Cl}^{\wedge}$ and $\mathrm{Cl}^{\vee\!\wedge}$ are defined by

$$\begin{aligned}
\mathrm{Cl}^{\top}_X\, \theta &= \theta \cup \mathsf{T} \\
\mathrm{Cl}^{\wedge}_X\, \theta &= \{\, \textstyle\bigwedge_{i=1}^{n} V_i \;:\; n \in \mathbb{N}, V_i \in \theta \,\} \\
\mathrm{Cl}^{\vee\!\wedge}_X\, \theta &= \left\{\, \textstyle\bigvee_{i=1}^{n} \bigwedge_{j=1}^{m} V_{ij} \;:\; n, m \in \mathbb{N}, V_{ij} \in \theta \,\right\}
\end{aligned}$$

It is straightforward to check that the above equations indeed define closures according to Definition 3.4.

Finally, sets of B-test constructors from Definition 4.1, together with suitably chosen closures, induce liftings of B to endofunctors on **TS** along the lines of Proposition 3.5:

Definition 4.4 For $\mathsf{W} \in \{\mathsf{Tr}, \mathsf{RdTr}\}$, the endofunctor on **TS** induced by the set of B-test constructors W and by the closure $\mathrm{Cl}^{\top}$ is denoted B^{W}. The endofunctor on **TS** induced by the set of B-test constructors Tr and by the closure $\mathrm{Cl}^{\wedge}$ is denoted B^{S}. The endofunctor on **TS** induced by the set of B-test constructors BS and by the closure $\mathrm{Cl}^{\vee\!\wedge}$ is denoted B^{BS}.

4.2 Relation to Modal Logics

It is not difficult to notice that the sets of test constructors shown in Definition 4.2 are related to modal constructors from the BNF grammars shown in Definition 2.2. This is not a coincidence, and indeed the modal logics from Definition 2.2 inspired Definitions 4.1-4.3. This subsection is devoted to giving a formal correspondence between these definitions.

Definition 4.5 Let $\mathsf{W} \in \{\mathsf{Tr}, \mathsf{RdTr}, \mathsf{S}, \mathsf{BS}\}$, and let $h : X \to BX$ be an LTS.

Define a function $[\![-]\!]_h : \mathcal{F}_{\mathsf{W}} \to (X \to 2)$ inductively as follows:

$$[\![\top]\!]_h = \mathsf{T} \qquad [\![\bot]\!]_h = \mathsf{F}$$

$$[\![\langle a \rangle\, \phi]\!]_h = w_{\langle a \rangle} \circ B[\![\phi]\!]_h \circ h \qquad [\![[a]\phi]\!]_h = w_{[a]} \circ B[\![\phi]\!]_h \circ h$$

$$[\![\check{Q} \wedge \langle a \rangle\, \phi]\!]_h = \check{w}_{aQ} \circ B[\![\phi]\!]_h \circ h$$

$$[\![\phi_1 \wedge \phi_2]\!]_h = [\![\phi_1]\!]_h \wedge [\![\phi_2]\!]_h \qquad [\![\phi_1 \vee \phi_2]\!]_h = [\![\phi_1]\!]_h \vee [\![\phi_2]\!]_h$$

The correspondence between modal formulae and tests is given in

Theorem 4.6 Let $\mathsf{W} \in \{\mathsf{Tr}, \mathsf{RdTr}, \mathsf{S}, \mathsf{BS}\}$ and let $h : X \to BX$ be an LTS. For any formula $\phi \in \mathcal{F}_{\mathsf{W}}$ and any $x \in X$,

$$[\![\phi]\!]_h x = \mathtt{tt} \iff x \models_h \phi$$

Proof Straightforward structural induction on modal formulae. □

The above correspondence maps modal logics to the least test suites lifting coalgebras to endofunctors B^{W}:

Theorem 4.7 Let $h : X \to BX$ be a B-coalgebra and $k : X \to \Omega$ the coinductive extension of h. For $\mathsf{W} \in \{\mathsf{Tr}, \mathsf{RdTr}, \mathsf{S}, \mathsf{BS}\}$,

$$\{[\![\phi]\!]_h \mid \phi \in \mathcal{F}_{\mathsf{W}}\} = k^* \omega^{\mathsf{W}}$$

where ω^{W} is taken from the final B^{W}-coalgebra $\phi : \langle \Omega, \omega^{\mathsf{W}} \rangle \to \langle B\Omega, B^{\mathsf{W}}_{\Omega} \omega^{\mathsf{W}} \rangle$.

Theorem 4.7 allows one to describe operational preorders and equivalences on a given coalgebra $h : X \to BX$ as specialization relations of the least test suites which lift h to coalgebras of various endofunctors on **TS**, as the following easy corollary shows.

Corollary 4.8 Let $h : X \to BX$ be a B-coalgebra and $k : X \to \Omega$ the coinductive extension of h. For $\mathsf{W} \in \{\mathsf{Tr}, \mathsf{RdTr}, \mathsf{S}, \mathsf{BS}\}$,

$$\sqsubseteq_{\mathsf{W}} = \leq_{k^* \omega^{\mathsf{W}}} \qquad \text{and} \qquad \cong_{\mathsf{W}} = \cong_{k^* \omega^{\mathsf{W}}}$$

Proof Compare Definitions 2.4 and 3.2 and use Theorems 4.6 and 4.7. □

This important corollary gives a coalgebraic characterization of preorders and equivalences from the van Glabbeek spectrum. However, an even stronger correspondence between B^{W}-coalgebras and operational relations can be shown, providing a full characterization of B^{W}-coalgebras. In Section 5 we present this correspondence, followed by an application of it to deriving coinductive proof principles in Section 6.

5 Characterization of Test Suite Coalgebras

We begin by characterizing the operational preorders and equivalences of B^{S} and B^{BS}-coalgebras. The remaining preorders and equivalences from the van Glabbeek spectrum are treated later.

The following two theorems show that the specialization preorders of B^{S}-coalgebras are exactly reflexive and transitive simulations (see Definition 2.5).

Theorem 5.1 For any B^{S}-coalgebra $h : \langle X, \theta\rangle \to \left\langle BX, B^{\mathsf{S}}_X\theta\right\rangle$, the specialization preorder $\leq_\theta$ is a reflexive and transitive simulation on $h : X \to BX$.

Theorem 5.2 For any coalgebra $h : X \to BX$ in **Set**, and any reflexive and transitive simulation R on h, there exists a test suite θ_R on X such that $\leq_{\theta_R} = R$ and $h : \langle X, \theta_R\rangle \to \left\langle BX, B^{\mathsf{S}}_X\theta_R\right\rangle$ is a valid B^{S}-coalgebra.

Proof (sketch) Assume any reflexive and transitive simulation R on $h : X \to BX$ and consider the following test suite on X:

$$\theta_R = \{\, V : X \to 2 \,:\, V \text{ is } R\text{-upper}\,\}$$

where V is R-upper means that for any $x, y \in X$, if xRy and $Vx = \mathtt{tt}$ then $Vy = \mathtt{tt}$. Then check that θ_R satisfies the above conditions. □

Corollary 5.3 Specialization preorders $\leq_\theta$ of B^{S}-coalgebras $h : \langle X, \theta\rangle \to \left\langle BX, B^{\mathsf{S}}_X\theta_R\right\rangle$ are exactly reflexive and transitive simulations on $h : X \to BX$. Specialization equivalences $\cong_\theta$ for B^{S}-coalgebras $h : \langle X, \theta\rangle \to \left\langle BX, B^{\mathsf{S}}_X\theta_R\right\rangle$ are exactly the equivalence relations $R \cap R^{-1}$ associated to reflexive and transitive simulations R on $h : X \to BX$.

A similar characterization of B^{BS}-coalgebras can be obtained in an analogous fashion:

Corollary 5.4 Specialization preorders $\leq_\theta$ of B^{BS}-coalgebras $h : \langle X, \theta\rangle \to \left\langle BX, B^{\mathsf{BS}}_X\theta_R\right\rangle$ are exactly reflexive and transitive bisimulations on $h : X \to BX$. Specialization equivalences $\cong_\theta$ for B^{BS}-coalgebras $h : \langle X, \theta\rangle \to \left\langle BX, B^{\mathsf{BS}}_X\theta_R\right\rangle$ are exactly the equivalence relations $R \cap R^{-1}$ associated to reflexive and transitive bisimulations R on $h : X \to BX$.

From Corollaries 5.4 and 4.8 it is easy to infer that the bisimulation preorder $\sqsubseteq_{\mathsf{BS}}$ on any $h : X \to BX$ is the largest bisimulation on h, which is a well-known result, stated in Proposition 2.9. This is related to the usual coinduction proof principle used to show that certain operations respect bisimulation equivalence. Many examples of proofs using this principle are shown, e.g., in [15, Section 12].

This hints that a similar characterization of B^{W}-coalgebras for various W might lead to analogous proof principles for other equivalences in the van Glabbeek spectrum. Such characterizations will be presented now, and an example of the resulting proof principle is shown in Section 6.

To give a full characterization of (the specialization relations $\leq_\theta$ and $\cong_\theta$ of) B^{W}-coalgebras for $\mathsf{W} \in \{\mathsf{Tr}, \mathsf{RdTr}\}$, we first need a few technical definitions and results.

Definition 5.5 A relation $S \subseteq X \times \mathcal{P}X$ is called a *quasi-preorder* on X if

- for any $x \in X$, $xS\{x\}$ and $x \not{S} \emptyset$,
- for any $x, y \in X$, $\xi, \chi \subseteq X$, if $xS\xi$, $y \in \xi$ and $yS\chi$ then $xS((\xi \setminus \{y\}) \cup \chi)$.

Definition 5.6 Let S be a quasi-preorder on X. A set $V \subseteq X$ is called *quasi-S-upper*, if for any $x \in V$, and for any $\xi \subseteq X$ such that $xS\xi$, the intersection $V \cap \xi$ is not empty.

Lemma 5.7 Let S be a quasi-preorder on X, and fix arbitrary elements $x, y \in X$. If for every quasi-S-upper set $Y \subseteq X$, $x \in Y$ implies $y \in Y$, then $xS\{y\}$.

Proof (sketch) Show that $Y = \{\, z \in X \;:\; z \not{S} \{y\} \,\}$ is quasi-S-upper. Since $y \notin Y$, also $x \notin Y$, hence $xS\{y\}$. □

Definition 5.8 Consider an LTS (a B-coalgebra) $h : X \to BX$. Let a range over A, Q over subsets of A, x, x' over X, and ξ over subsets of X. A relation $S \subseteq X \times \mathcal{P}X$ is called a

- *one-by-many* Tr*-simulation* on h if whenever $xS\xi$ and $x \xrightarrow{a} x'$, then $x'S\{y' \in X \mid \exists y \in \xi.\ y \xrightarrow{a} y'\}$, *one-by-many* RdTr*-simulation* on h if S is a one-by-many Tr-simulation on h, and if whenever $xS\xi$ and $x \xrightarrow{a} x'$, then $x'S\left\{\, y' \in X \;:\; \exists y \in \xi.\ y \xrightarrow{a} y', I(x) = I(y) \,\right\}$.

For $\mathsf{W} \in \{\mathsf{Tr}, \mathsf{RdTr}\}$, a relation $R \subseteq X \times X$ is called W*-aware* on h if there exists a one-by-many W-simulation S on h such that $xRy \iff xS\{y\}$. If, moreover, S is a quasi-preorder, then R is called a W*-aware preorder*.

The following theorems and corollaries characterize the specialization preorders and equivalences of B^{W}-coalgebras as respective W-aware preorders.

Theorem 5.9 Let $\mathsf{W} = \{\mathsf{Tr}, \mathsf{RdTr}\}$. For any B^{W}-coalgebra $h : \langle X, \theta \rangle \to \left\langle BX, B^{\mathsf{W}}_X \theta \right\rangle$, the relation $\leq_\theta$ is a W-aware preorder on $h : X \to BX$.

Proof We prove the case $\mathsf{W} = \mathsf{Tr}$, the other case is similar. First, recall that saying that $h : \langle X, \theta \rangle \to \left\langle BX, B^{\mathsf{Tr}}_X \theta \right\rangle$ is a B^{Tr}-coalgebra is equivalent to saying that $\theta \supseteq h^* B^{\mathsf{Tr}}_X \theta$. Equivalently, $\mathsf{T} \in \theta$ and for any $V \in \theta$ and $a \in A$, also

$w_{\langle a \rangle} \circ BV \circ h \in \theta$. Define $S \subseteq X \times \mathcal{P}X$ as follows:

$$xS\xi \text{ iff } \forall V \in \theta.\ (Vx = \mathtt{tt} \Rightarrow \exists y \in \xi.\ Vy = \mathtt{tt})$$

Obviously $x \leq_\theta y$ if and only if $xS\{y\}$, hence it is enough to show that S is a quasi-preorder and a one-by-many Tr-simulation.

The proof that S is a quasi-preorder is straightforward. To show that it is a one-by-many Tr-simulation, consider any $x \in X$, $\xi \subseteq X$. Assume $x \xrightarrow{a} x'$ and $x' \not S \{y' \in X \mid \exists y \in \xi.\ y \xrightarrow{a} y'\}$. By definition of S, there exists a test $V \in \theta$ such that

- $Vx' = \mathtt{tt}$, and
- $Vy' = \mathtt{ff}$ for all $y \in \xi$, $y \xrightarrow{a} y'$.

Consider a test $V' = w_{\langle a \rangle} \circ BV \circ h$. Obviously $V'x = \mathtt{tt}$, but for every $y \in \xi$, $V'y = \mathtt{ff}$. Since $V \in \theta$, also $V' \in \theta$, hence $x \not S \xi$. □

Theorem 5.10 Let $\mathsf{W} = \{\mathsf{Tr}, \mathsf{RdTr}\}$. For any coalgebra $h : X \to BX$ in **Set**, and any W-aware preorder R on h, there exists a test suite θ_R on X such that $\leq_{\theta_R} = R$ and $h : \langle X, \theta_R \rangle \to \left\langle BX, B^{\mathsf{W}}_X \theta_R \right\rangle$ is a valid B^{W}-coalgebra.

Proof Again, we prove the case of $\mathsf{W} = \mathsf{Tr}$. The other case is similar. Assume a quasi-preorder and one-by-many Tr-simulation S on h such that xRy if and only if $xS\{y\}$. Define (compare Theorem 5.2)

$$\theta_R = \{V : X \to 2 \mid V \text{ is quasi-}S\text{-upper}\}$$

To check that $R \subseteq \leq_{\theta_R}$, assume xRy, or equivalently, $xS\{y\}$. Consider any $V \in \theta_R$ such that $Vx = \mathtt{tt}$. Since V is quasi-S-upper, also $Vy = \mathtt{tt}$.

To check that $\leq_{\theta_R} \subseteq R$, assume that for every $V : X \to 2$ such that V is quasi-S-upper, if $Vx = \mathtt{tt}$ then $Vy = \mathtt{tt}$. By Lemma 5.7 it easily follows that $xS\{y\}$, hence xRy.

To check that $h : \langle X, \theta \rangle \to \left\langle BX, B^{\mathsf{Tr}}_X \theta_R \right\rangle$ is a valid B^{Tr}-coalgebra, it is enough to check that

- $\mathsf{T} \in \theta_R$, and
- for any $V \in \theta_R$, also $w_{\langle a \rangle} \circ BV \circ h \in \theta_R$.

The first condition is easy, since T is quasi-S-upper for any quasi-preorder S on X.

For the second condition, assume any $V \in \theta_R$ and denote $V' = w_{\langle a \rangle} \circ BV \circ h$. Take any $x \in X$, $\xi \subseteq X$ such that $xS\xi$ and $V'x = \mathtt{tt}$. The latter assumption means that there exists an $x' \in X$ such that $x \xrightarrow{a} x'$ and $Vx' = \mathtt{tt}$.

Since S is a one-by-many Tr-simulation, this means that $x'S\{y' \in X \mid$

$\exists y \in \xi.\ y \xrightarrow{a} y'\}$ and (since V is quasi-S-upper) there exist $y, y' \in X$ such that $y \in \xi$, $y \xrightarrow{a} y'$ and $Vy' = \mathtt{tt}$. Then $V'y = \mathtt{tt}$. Since $xS\xi$ were chosen arbitrarily, V' is quasi-S-upper. $\square$

Corollary 5.11 Specialization preorders $\leq_\theta$ for B^{Tr}-coalgebras $h : \langle X, \theta\rangle \to \left\langle BX, B^{\mathsf{Tr}}_X \theta_R \right\rangle$ are exactly Tr-aware preorders on $h : X \to BX$.

The trace preorder $\sqsubseteq_{\mathsf{Tr}}$ on h is the largest Tr-aware relation on h.

6 Application: coinduction principle for traces

One of the most useful applications of coalgebraic semantics of processes is the coinduction proof principle, based on the fact that the bisimulation equivalence on any LTS is the largest bisimulation on it. Therefore to prove that two processes are bisimulation equivalent, it is enough to provide any bisimulation that relates them. The rich structure of bisimulation relations allows one to use this proof principle in a very convenient fashion. Many examples of its use can be found e.g. in [15, Section 12].

Results shown in Section 5 allow one to apply similar reasoning to other equivalences from the van Glabbeek spectrum. Corollary 5.11 characterizes trace preorder as the largest Tr-aware relation. As it turns out, Tr-aware relations have enough structure to play in reasoning about trace equivalence a similar rôle to that of bisimulations in reasoning about bisimulation equivalence. We now show an example of such "coinduction principle for traces".

Consider a final B-coalgebra $\phi : \Omega \to B\Omega$ (recall Proposition 2.9). On the set Ω, define the associative, idempotent and commutative binary operation $+$ by

$$p + q = \phi^{-1}(\phi p \cup \phi q)$$

Now define a function $\mathtt{glue} : \Omega \to \Omega$ as the coinductive extension of the B-coalgebra $\alpha : \Omega \to B\Omega$ defined by

$$\alpha p = \left\{ \left\langle a, \textstyle\coprod_{\langle a,p'\rangle \in \phi p} p' \right\rangle \,:\, a \in A \right\}$$

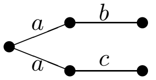
where $\coprod$ denotes the obvious extension of $+$ to finite subsets of Ω.

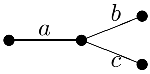
Using the "operational rule" notation for coinductive definitions as introduced in [15, Section 11], one may write alternatively

$$\frac{p_1, \ldots, p_n \text{ are exactly the processes for which } p \xrightarrow{a} p_i}{\mathtt{glue}(p) \xrightarrow{a} \mathtt{glue}(p_1 + \cdots + p_n)}$$

for any $a \in A$.

For example, if Ω is the set of all finitely branching labelled synchronisation trees, one has

`glue(` 

`) =` 

Theorem 6.1 The operation `glue` preserves and respects traces. In other words, for any process $p \in \Omega$, p and $\mathtt{glue}(p)$ are trace equivalent.

Proof A standard way of proving this theorem is to use induction on the length of traces. Instead, we use the coinduction proof principle for traces, as expressed in Corollary 5.11.

First, we show that the relation $\{ \langle \mathtt{glue}(p), p \rangle \; : \; p \in \Omega \}$ is a Tr-aware relation. Consider $S \subseteq \Omega \times \mathcal{P}\Omega$ defined by

$$pS\{q_1, \ldots, q_n\} \iff \mathtt{glue}\left(\textstyle\coprod_{i \in I} q_i\right) = p \text{ for some } I \subseteq \{1, \ldots, n\}.$$

To show that S is a one-by-many Tr-simulation, consider any $a \in A$, $p, p' \in \Omega$, $\xi \in \Omega$ such that $pS\xi$ and $p \xrightarrow{a} p'$. By definition of S, $p = \mathtt{glue}\left(\coprod_{i \in I} q_i\right)$ for some index set I such that $q_i \in \xi$ for every $i \in I$. By definitions of `glue` and $\coprod$, there is

$$p' = \mathtt{glue}\left(\textstyle\coprod_{i \in I,\ q_i \xrightarrow{a} q_i'} q_i'\right)$$

hence

$$p'S\left\{ q' \in \Omega \; : \; \exists q \in \xi.\ q \xrightarrow{a} q' \right\}$$

This concludes the first part of the proof.

Next, we show that the relation $\{ \langle p, \mathtt{glue}(p) \rangle \; : \; p \in \Omega \}$ is contained in a Tr-aware relation. Consider $S \subseteq \Omega \times \mathcal{P}\Omega$ defined by

$$pS\xi \iff \mathtt{glue}(p + q) \in \xi \text{ for some } q \in \Omega.$$

To show that S is a one-by-many Tr-simulation, consider any $a \in A$, $p, p' \in \Omega$, $\xi \in \Omega$ such that $pS\xi$ and $p \xrightarrow{a} p'$. This means that also $p+q \xrightarrow{a} p'$. Then, by definition of `glue`, $\mathtt{glue}(p + q) \xrightarrow{a} \mathtt{glue}(p' + q')$ for some $q' \in \Omega$. From this it follows that

$$\mathtt{glue}(p' + q') \in \left\{ r' \in \Omega \; : \; \exists r \in \xi.\ r \xrightarrow{a} r' \right\}$$

hence

$$p'S\left\{ r' \in \Omega \; : \; \exists r \in \xi.\ r \xrightarrow{a} r' \right\}$$

and S is a one-by-many simulation, which concludes the proof. $\square$

7 Related work

The abstract test suite approach to process equivalence is novel, but several other approaches to the abstract representation of operational equivalences are known. The most popular coalgebraic approach to bisimulation, based on coalgebra spans [1,15], focuses on a single, canonical notion of equivalence for every notion of behaviour. In the case of labelled transition system, this abstract notion specializes to bisimulation equivalence. A few attempts have been made to modify the coalgebra span approach to cover trace equivalence. In [14,16], the underlying category was changed to that of semilattices. In [13], instead, the notion of coalgebra morphism was changed. Both approaches led to the definition of trace equivalence as the (span) bisimulation in an appropriate category.

The approach taken in [13] bears some similarities to the test suite framework. In both approaches, the coalgebras modelling transition systems are lifted to another category. However, these liftings seem fundamentally different. In [13], the lifting is based on a distributive law of the functor for deterministic behaviour over the powerset monad, thus exploiting the structure of the behaviour endofunctor for LTSs. Also, in the resulting category the coalgebras in question remain unchanged, but the notion of coalgebra morphism is changed. In our abstract approach, the structure of the behaviour functor is not used in any obvious way, and indeed we never use the fact that the powerset functor is a monad. Also, in the category of B^{Tr}-coalgebras the coalgebra morphisms come from the underlying category of B-coalgebras, but the coalgebras themselves are different (they are equipped with test suites and required to be prefixed points of certain operators).

Our approach has much stronger connections to that of [4,6], where behaviour endofunctors are canonically lifted to the category **Rel** of binary relations and relation preserving functions, which is fibred over **Set**. A similar technique was used in [12] in the context of recursively defined domains, where even a counterpart of our operator Φ_h^{W} was defined. In the test suite approach, behaviour endofunctors are lifted to the category **TS**, which is also fibred over **Set**, but has more structure than **Rel** and thus allows one to represent e.g., trace equivalence. Also, in the test suite approach we resign from the canonicity of liftings (indeed, arbitrary sets of test constructors can be used), thus allowing one to represent more notions of operational equivalence.

All the abstract approaches mentioned above aim to define abstract notions of particular equivalences, parametrized by the notion of behaviour: bisimulation equivalence [15,4], simulation equivalence [6], or trace equivalence [14,13]. Our goal is, in a sense, more modest; we aim at an abstract and general definition of a well-behaved process equivalence. This allows us to cover many more examples than other approaches, but prevents us from formalizing a connection between, e.g., the notions of bisimulation for different notions of behaviour.

Related to modal logics, the test suite approach provides an abstract notion of modal operator as a test suite constructor. It must be investigated how this notion is related to that of natural relation, used to represent modal operators in the coalgebraic logic [10,11].

It must be noted that three equivalences from the van Glabbeek spectrum: possible futures, possible worlds and 2-nested simulation equivalence cannot be described in the test suite framework shown in this paper. This is related to the fact that the modal formulae describing these equivalences cannot be presented by a grammar with only one nonterminal. This problem can be circumvented by choosing a different set of test values instead of 2 (see [7]).

References

[1] P. Aczel and N. Mendler. A final coalgebra theorem. In D. H. Pitt et al., editors, *Proc. CTCS'89*, volume 389 of *LNCS*, pages 357–365, 1989.

[2] M. Barr. Terminal coalgebras for endofunctors on sets. *Theoretical Computer Science*, 114:299–315, 1993.

[3] R. J. van Glabbeek. The linear time - branching time spectrum I. In J. Bergstra, A. Ponse, and S. Smolka, editors, *Handbook of Process Algebra*. Elsevier, 1999.

[4] C. Hermida and B. Jacobs. Structural induction and coinduction in a fibrational setting. *Information and Computation*, 145(2):107–152, 1998.

[5] B. Jacobs. *Categorical Logic and Type Theory*. Number 141 in Studies in Logic and the Foundations of Mathematics. North Holland, Amsterdam, 1999.

[6] B. Jacobs and J. Hughes. Simulations in coalgebra. In H. Peter Gumm, editor, *Electronic Notes in Theoretical Computer Science*, volume 82. Elsevier, 2003.

[7] B. Klin. *An abstract approach to process equivalence for well-behaved operational semantics*. PhD dissertation, BRICS, Aarhus University, 2004. Forthcoming.

[8] B. Klin and P. Sobocinski. Syntactic formats for free: An abstract approach to process equivalence. In R. Amadio and D. Lugiez, editors, *Proc. CONCUR 2003*, volume 2671 of *LNCS*, 2003.

[9] B. Klin and P. Sobocinski. Syntactic formats for free: An abstract approach to process equivalence. BRICS Report RS-03-18, Aarhus University, 2003.

[10] L. Moss. Coalgebraic logic. *Annals of Pure and Applied Logic*, 96:177–317, 1999.

[11] D. Pattinson. Semantical principles in the modal logic of coalgebras. In *Proc. STACS 2001*, volume 2010 of *LNCS*. Springer Verlag, 2001.

[12] A. M. Pitts. A co-induction principle for recursively defined domains. *Theoretical Computer Science*, 124(2):195–219, 1994.

[13] J. Power and D. Turi. A coalgebraic foundation for linear time semantics. *Electronic Notes in Theoretical Computer Science*, 29, 1999.

[14] J. Rutten and D. Turi. Initial algebra and final coalgebra semantics for concurrency. In J. de Bakker et al., editors, *Proc. of the REX workshop A Decade of Concurrency – Reflections and Perspectives*, volume 803 of *LNCS*, pages 530–582. Springer-Verlag, 1994.

[15] J. J. M. M. Rutten. Universal coalgebra: a theory of systems. *Theoretical Computer Science*, 249:3–80, 2000.

[16] D. Turi. Categorical modeling of structural operational rules: case studies. In *Proc. CTCS'97*, volume 1290 of *LNCS*, pages 127–146. Springer-Verlag, 1997.

[17] D. Turi and G. Plotkin. Towards a mathematical operational semantics. In *Proceedings 12th Ann. IEEE Symp. on Logic in Computer Science, LICS'97, Warsaw, Poland, 29 June – 2 July 1997*, pages 280–291. IEEE Computer Society Press, 1997.