On Linear Contravariant Semantics*

Ignacio Fábregas, David de Frutos Escrig, and Miguel Palomino

Departamento de Sistemas Informáticos y Computación, UCM
fabregas@fdi.ucm.es {defrutos, miguelpt}@sip.ucm.es

Abstract. Covariant-contravariant simulation and conformance simulation generalize plain simulation and aim at capturing the fact that it is not always the case that "the larger the number of behaviors, the better". We have already studied in detail their corresponding simulation semantics, which are located in the branching-time side of the generalized ltbt-spectrum. In this paper we concentrate on the linear-time side of the spectrum, starting with the development of the adequate notions of covariant-contravariant and conformance traces. Then, we continue by introducing suitable versions of the failures semantics and the rest of the other classic linear semantics in the case of conformance semantics.

1 Introduction and motivation

Simulations are a very natural way to compare systems defined by labeled transition systems or other related mechanisms based on describing the behavior of states by means of the actions they can execute. However, the classic notion of simulation does not take into account the fact that whenever a system has several possibilities for the execution of an action, it will choose in an unpredictable manner, resulting in more non-determinism and less control.

We have proposed two new simulation notions which are more suitable to deal with non-determinism [6]. On the one hand, covariant-contravariant simulations were designed to manage systems in which non-determinism arises because of the presence of both input and output actions; on the other hand, conformance simulations cope with having several options for the same action. In previous works we have proved that these simulations can be presented as instances of the coalgebraic simulation framework [6] and have also described their logical characterizations [8].

In this paper we continue with the study of these two frameworks, this time by concentrating on the linear-time side of the spectrum, starting with the development of the adequate notions of covariant-contravariant and conformance traces. The most used linear semantics are located in the ltbl [12, 5] just below the ready simulation semantics and therefore, before studying their variations, it is natural to extend ready simulation to the new framework. Thus, we develop the adequate notion of covariant-contravariant failures, whereas for the conformance framework we show that the suitable versions of all the classic linear semantics are just the "opposite" of the classic notions. This is so because the ready conformance simulation preorder turns out to be the inverse of the

* Research supported by the Spanish projects DESAFIOS10 TIN2009-14599-C03-01, TESIS TIN2009-14321-C02-01 and PROMETIDOS S2009/TIC-1465.

plain ready simulation preorder which, to some extent, proves the adequacy of all these original semantics also in the conformance framework.

In our previous works on the subject the relation between covariant-contravariant semantics and those for I/O systems [11], which have been more recently integrated in the study of interface I/O automata [9], has been noted. We plan to consider these recent developments in order to look for applications of our new covariant-contravariant semantics. The same is true for the modal transition systems which have been related to the covariant-contravariant systems in [3].

Regarding conformance semantics, the first related references are also quite old [10] and correspond to the notion of conformance testing, which is close to failures semantics. Our simple results supporting the use of the classic linear semantics in the conformance framework prove that no new definitions are needed to translate those of the classic linear semantics to the new setting. As was done in [4], where characteristic formulas were built for our conformance simulation semantics, we expect to be able to complete the transfer of all the results on the classic semantics to the conformance framework, finding new applications supported by the consideration of its very foundations.

All the missing proofs can be found in the extended version at `http://maude.sip.ucm.es/~miguelpt/bibliography.html`.

2 Preliminaries

In this section we summarize, for the sake of completeness and easy reference, some definitions and concepts from [12, 6, 7]. First, let us recall that the set *BCCSP(A)* of basic processes for the alphabet A is defined by the BNF-grammar

$$p ::= 0 \mid ap \mid p + q$$

where $a \in A$. With these operators we can only define finite processes; however, it is well known that these operators capture the essence of any transition system which can be defined by a (possibly infinite) system of equations specifying the behavior of each state.

Next, let us recall our two new simulation notions and their axiomatizations.

Definition 1. *Given $P = (P, A, \rightarrow_P)$ and $Q = (Q, A, \rightarrow_Q)$, two labeled transition systems for the alphabet A, and $\{A^r, A^l\}$[1] a partition of this alphabet, a (A^r, A^l)-**simulation** (or just a **covariant-contravariant simulation**) between them is a relation $R \subseteq P \times Q$ such that for every pRq we have:*

- *For all $a \in A^r$ and all $p \xrightarrow{a} p'$ there exists $q \xrightarrow{a} q'$ with $p'Rq'$.*
- *For all $b \in A^l$ and all $q \xrightarrow{b} q'$ there exists $p \xrightarrow{b} p'$ with $p'Rq'$.*

We will write $p \lesssim_{cc} q$ if there exists a covariant-contravariant simulation R such that pRq.

[1] In [6, 8, 7] we used a more general definition which also includes bivariant actions but, since we have noticed that in the presence of those actions some technical difficulties appear, we prefer to omit them in this presentation.

Remark 1 (Notation). In the following we will use a to denote the generic actions in A^r and b to denote those in A^l.

Definition 2. *Given $P = (P, A, \rightarrow_P)$ and $Q = (Q, A, \rightarrow_Q)$ two labeled transition systems for the alphabet A, a* ***conformance simulation*** *between them is a relation $R \subseteq P \times Q$ such that whenever pRq, then:*

- *For all $a \in A$, if $p \xrightarrow{a}$ then $q \xrightarrow{a}$ (this means, using the usual notation for process algebras, that $I(p) \subseteq I(q)$).*
- *For all $a \in A$ such that $q \xrightarrow{a} q'$ and $p \xrightarrow{a}$, there exists some p' with $p \xrightarrow{a} p'$ and $p'Rq'$ (this means that whenever $a \in I(p) \cap I(q)$, R behaves locally as a "contravariant" simulation).*

We will write $p \lesssim_{cf} q$ if there exists a conformance simulation R such that pRq.

Let us recall the definitions in [12] of the most classic linear semantics for (plain) processes.

Definition 3 (Trace semantics). *We say that $\alpha = a_1 \ldots a_n$, with $0 \leq n$ is a trace of the process p (denoted by $\alpha \in \mathrm{Tr}(p)$) if there exist processes p_i with $i \in \{1, \ldots n\}$ such that $p \xrightarrow{a_1} p_1 \xrightarrow{a_2} p_2 \ldots \xrightarrow{a_n} p_n$. We also write $p \xrightarrow{\alpha} p_n$.*

We will denote by $\mathrm{Tr}(p)$ the set of traces of the process p, and write $p \lesssim_T q$ if $\mathrm{Tr}(p) \subseteq \mathrm{Tr}(q)$.

Definition 4 (Failure semantics). *We say that $\langle \alpha, X \rangle$ is a failure pair of a process p (denoted by $\langle \alpha, X \rangle \in \mathrm{F}(p)$) if there exists p' such that $p \xrightarrow{\alpha} p'$ and $I(p') \cap X = \emptyset$. We will denote $\mathrm{F}(p)$ the set of failures of the process p, and write $p \lesssim_F q$ if $\mathrm{F}(p) \subseteq \mathrm{F}(q)$.*

Definition 5 (Ready simulation semantics). *We say that $R \subseteq P \times Q$ is a ready simulation if, whenever pRq and $a \in A$, we have*

- *$I(p) = I(q)$, and*
- *if $p \xrightarrow{a} p'$, then there exists $q \xrightarrow{a} q'$ such that $p'Rq'$.*

We will write $p \lesssim_{RS} q$ if there exists a ready simulation R such that pRq.

3 Trace semantics for covariant-contravariant processes

Plain traces (covariant traces for us in the following) associate to each process p a set of traces $\mathrm{Tr}(p)$, in such a way that the trace preorder $\lesssim_T$ is defined by simply taking $p \lesssim_T q$ iff $\mathrm{Tr}(p)) \subseteq \mathrm{Tr}(q)$. Instead, whenever all the actions are contravariant, contravariant traces should allow us to define just the "opposite"[2] order $p \lesssim_{cT} q$ iff $q \lesssim_T p$, by means of set inclusion between the corresponding sets of contravariant traces $\mathrm{Tr}_C(p)$; that is, $p \lesssim_{cT} q$ iff $\mathrm{Tr}_C(p) \subseteq \mathrm{Tr}_C(q)$. Since obviously we have $T_1 \subseteq T_2$ iff $\overline{T_2} \subseteq \overline{T_1}$, in this simple case it would be sufficient to take as contravariant traces the set of *no-traces* of each process, that is, the complement of the set of traces: $\mathrm{Tr}_C(p) = \overline{\mathrm{Tr}(p)}$.

[2] We have included the quotation marks because we are changing covariant by contravariant actions and therefore, if we fix a partition of the set of actions $\{A^r, A^l\}$, then we should substitute the corresponding covariant actions by some "paired" contravariant ones in order to obtain the corresponding contravariant "image" of the original covariant process.

Definition 6. *We say that $\alpha = b_1 \dots b_n$ is a contravariant trace (or a "no-trace") of the process p (denoted by $\alpha \in \mathrm{Tr}_C(p)$) if there is no sequence $p \xrightarrow{b_1} p_1 \xrightarrow{b_2} p_2 \dots \xrightarrow{b_n} p_n$, for any processes p_i.*

Proposition 1. *Contravariant traces have the following properties:*

- *$\langle\rangle \notin \mathrm{Tr}_C(p)$ for any p.*
- *Given $\alpha = b_1 \dots b_{n-1}$, $\alpha \cdot \langle b_n \rangle \in \mathrm{Tr}_C(p)$ if and only if either $\alpha \in \mathrm{Tr}_C(p)$, or $\alpha \in \mathrm{Tr}(p)$ and for all $p \overset{b_1 \dots b_{n-1}}{\longrightarrow} p_{n-1} \overset{b_n}{\not\longrightarrow}$. As a consequence, any set of contravariant traces is closed by extensions, that is, any extension of a contravariant trace is also a contravariant trace.*

Note the (expected!) duality with respect to the corresponding characterization of the set of covariant traces of a process.

Proof. First, it is trivial that $\langle\rangle \notin \mathrm{Tr}_C(p)$ for any p, since $p \xrightarrow{\langle\rangle} p$. On the other hand, let $\alpha = b_1 \dots b_{n-1}$ and $\alpha \cdot \langle b_n \rangle \in \mathrm{Tr}_C(p)$: by definition, it is immediate to check that either $\alpha \in \mathrm{Tr}_C(p)$, or $p \xrightarrow{b_1} p_1 \xrightarrow{b_2} p_2 \dots \xrightarrow{b_{n-1}} p_{n-1}$ and for all $p_{n-1} \overset{b_n}{\not\longrightarrow}$. Analogously, if either $\alpha \in \mathrm{Tr}_C(p)$, or $\alpha \in \mathrm{Tr}(p)$ and for all $p \overset{b_1 \dots b_{n-1}}{\longrightarrow} p_{n-1} \overset{b_n}{\not\longrightarrow}$, it is straightforward to see that there is no sequence $p \xrightarrow{b_1} p_1 \xrightarrow{b_2} p_2 \dots \xrightarrow{b_{n-1}} p_{n-1} \xrightarrow{b_n} p_n$ for any chain of processes, that is, $\alpha \cdot \langle b_n \rangle \in \mathrm{Tr}_C(p)$. $\square$

Therefore, both covariant and contravariant traces allow us to compare processes by their linear computations considering:

- either the "optimistic" point of view: to execute actions is good, and therefore the more (covariant) traces, the better;
- or the "pessimistic" point of view: actions are all of them bad, therefore the less (covariant) traces (or the more contravariant traces) the better.

This corresponds to the input/output scenario where reactive and generative behaviours are mixed together: a server that accepts more calls is certainly more flexible, but instead when it produces its reply we expect a certain kind of output and in this case a more reduced set of outputs implies that we control the behaviour of the server in a strange way, and this means that the user is more simple and feasible, since it needs not to take care of so many scenarios.

However, in both cases plain set inclusion characterizes the corresponding order. It is clear that the set inclusion order captures the goal "the more possible behaviours (covariant traces) the better", but in the contravariant case we need to reflect the opposite situation, and this is the way contravariant traces correspond to (what would be) covariant no-traces.

Next, let us consider the general and more interesting case in which we have together both *good* actions in A^r and *bad* actions in A^l. By combining the two definitions above we obtain the following definition of covariant-contravariant traces.

Definition 7. *Given a partition $\{A^r, A^l\}$ of a set of actions, we say that α is a covariant-contravariant trace of the process p (denoted by $\alpha \in \mathrm{Tr}_{cc}(p)$) if and only if:*

– $\alpha = \langle\rangle$, *or*
– $\alpha = \langle a\rangle \in A^r$ *and* $p \xrightarrow{a}$, *or*
– $\alpha = \langle b\rangle$ *with* $b \in A^l$, *and* $p \not\xrightarrow{b}$, *or*
– $\alpha = \langle a\rangle \cdot \alpha'$, $a \in A^r$, *and* $p \xrightarrow{a} p'$ *with* $\alpha' \in \mathrm{Tr}_{cc}(p')$, *or*
– $\alpha = \langle b\rangle \cdot \alpha'$ *with* $b \in A^l$, $\alpha' \neq \langle\rangle$ *and* $\alpha' \in \mathrm{Tr}_{cc}(p')$ *for all* $p \xrightarrow{b} p'$.

The covariant-contravariant trace preorder $p \lesssim_{ccT} q$ *is defined by:* $p \lesssim_{ccT} q$ *iff* $\mathrm{Tr}_{cc}(p) \subseteq \mathrm{Tr}_{cc}(q)$.

It is easy to check that this definition nearly generalizes both the definition of plain traces and that of contravariant traces, which corresponds to the case $A^r = \emptyset$. We said "nearly" because when considering both covariant and contravariant actions together we need a uniform treatment of the empty sequence. However, since the semantics does not distinguish two different empty traces, we have to decide if we want $\langle\rangle \in \mathrm{Tr}_{cc}(p)$ for any process p, or $\langle\rangle \notin \mathrm{Tr}_{cc}(p)$. We preferred the first option in order to totally preserve the pure covariant traces when $A^l = \emptyset$. Moreover, once we remove the empty trace from the set of traces, we also obtain a generalization of the notion of pure contravariant traces for the case $A^r = \emptyset$.

As usual, $\mathrm{Tr}_{cc}(p)$ is the smallest set of sequences of actions in A defined by the clauses above. As a matter of fact, the second clause can also be obtained as particular case of the fourth, but we preferred to maintain it in the definition in order to emphasize the basic cases. Note that, instead, the third clause is not a particular case of the fifth, in order to reflect the asymmetric treatment of the empty trace in the contravariant case.

Example 1. Let us compute the covariant-contravariant traces of the process $p = b \cdot b \cdot a \cdot 0$.

– $\langle\rangle \in \mathrm{Tr}_{cc}(p)$, trivially.
– $\langle b\rangle \notin \mathrm{Tr}_{cc}(p)$, because neither the third clause (because $p \xrightarrow{b}$), nor the fifth clause are applicable. However, by the third clause $\langle b'\rangle \in \mathrm{Tr}_{cc}(p)$ for any $b' \in A^l$, $b' \neq b$. We also have $\langle b'\rangle \cdot \alpha \in \mathrm{Tr}_{cc}(p)$ for any $b' \in A^l$, $b' \neq b$ and $\alpha \in A^*$.
– Analogously to the previous case, we can see that $\langle bb\rangle \notin \mathrm{Tr}_{cc}(p)$, but $\langle bb'\rangle \cdot \alpha \in \mathrm{Tr}_{cc}(p)$ for all $b' \in A^l$, $b' \neq b$ and $\alpha \in A^*$.
– $\langle bba\rangle \in \mathrm{Tr}_{cc}(p)$, because $b \cdot b \cdot a \cdot 0 \xrightarrow{b} b \cdot a \cdot 0 \xrightarrow{b} a \cdot 0 \xrightarrow{a} 0$, by clauses five and two. Hence, also $\langle bbab'\rangle \cdot \alpha \in \mathrm{Tr}_{cc}(p)$ for all $b' \in A^l$ and $\alpha \in A^*$.
– No more traces belongs to $\mathrm{Tr}_{cc}(p)$.

It is well-known that classic traces are closed by prefixes, that is, if $\alpha \cdot \beta \in \mathrm{Tr}(p)$ then $\alpha \in \mathrm{Tr}(p)$. For covariant-contravariant traces the same is true for covariant actions. The following proposition makes this situation precise.

Proposition 2. *If* $\alpha' \cdot \langle a\rangle \cdot \beta \in \mathrm{Tr}_{cc}(p)$ *then,* $\alpha' \cdot \langle a\rangle \in \mathrm{Tr}_{cc}(p)$.

Proof. Let us prove that $\alpha' \cdot \langle a\rangle \in \mathrm{Tr}_{cc}(p)$ by structural induction on α'.

– $\alpha' = \langle\rangle$. By definition, if $\langle a\rangle \cdot \beta \in \mathrm{Tr}_{cc}(p)$, then $p \xrightarrow{a} p'$ and $\beta \in \mathrm{Tr}_{cc}(p')$. Hence, we also have that $\langle a\rangle \in \mathrm{Tr}_{cc}(p)$.

– $\alpha' = \langle b \rangle$. By definition, since $\langle a \rangle \cdot \beta \neq \langle\rangle$ and $\langle b \rangle \cdot \langle a \rangle \cdot \beta \in \mathrm{Tr}_{cc}(p)$ we have that $\langle a \rangle \cdot \beta \in \mathrm{Tr}_{cc}(p')$ for any $p \xrightarrow{b} p'$. Thus, also $\langle a \rangle \in \mathrm{Tr}_{cc}(p')$ for any $p \xrightarrow{b} p'$ and hence $\alpha' \cdot \langle a \rangle \in \mathrm{Tr}_{cc}(p)$.
– $\alpha' = \langle a \rangle \cdot \alpha''$. By definition, if $\langle a \rangle \cdot \alpha'' \cdot \langle a \rangle \cdot \beta \in \mathrm{Tr}_{cc}(p)$ then $p \xrightarrow{a} p'$ with $\alpha'' \cdot \langle a \rangle \cdot \beta \in \mathrm{Tr}_{cc}(p')$. Thus, by induction hypothesis, also $\alpha'' \cdot \langle a \rangle \in \mathrm{Tr}_{cc}(p')$, and hence $\alpha' \cdot \langle a \rangle \in \mathrm{Tr}_{cc}(p)$.
– $\alpha' = \langle b \rangle \cdot \alpha''$. This case is analogous to the previous one. □

For contravariant traces we have the dual result, as the following proposition makes precise.

Proposition 3. *If* $\alpha' \cdot \langle b \rangle \in \mathrm{Tr}_{cc}(p)$ *then, also* $\alpha' \cdot \langle b \rangle \cdot \beta \in \mathrm{Tr}_{cc}(p)$*, for all* β*.*

Proof. Indeed, let $\alpha' \cdot \langle b \rangle \in \mathrm{Tr}_{cc}(p)$ and let us see that $\alpha' \cdot \langle b \rangle \cdot \beta \in \mathrm{Tr}_{cc}(p)$ for all $\beta \in A^*$. The proof will be done by structural induction over α'.

– $\alpha' = \langle\rangle$. By definition, if $\langle b \rangle \in \mathrm{Tr}_{cc}(p)$, then $p \stackrel{b}{\not\longrightarrow}$. Hence, by the fifth clause, also $\langle b \rangle \cdot \beta \in \mathrm{Tr}_{cc}(p)$, for all $\beta \in A^*$.
– $\alpha' = \langle b \rangle$. By definition, since $\langle b \rangle \neq \langle\rangle$ and $\langle b \rangle \cdot \langle b \rangle \in \mathrm{Tr}_{cc}(p)$ we have that $\langle b \rangle \in \mathrm{Tr}_{cc}(p')$ for any $p \xrightarrow{b} p'$. Thus, we have that $p' \stackrel{b}{\not\longrightarrow}$ for all $p \xrightarrow{b} p'$ and hence $\langle b \rangle \cdot \beta \in \mathrm{Tr}_{cc}(p')$. That is, $\alpha' \cdot \langle b \rangle \cdot \beta \in \mathrm{Tr}_{cc}(p)$.
– $\alpha' = \langle a \rangle \cdot \alpha''$. By definition, if $\langle a \rangle \cdot \alpha'' \cdot \langle b \rangle \in \mathrm{Tr}_{cc}(p)$ then $p \xrightarrow{a} p'$ with $\alpha'' \cdot \langle b \rangle \in \mathrm{Tr}_{cc}(p')$. Thus, by induction hypothesis, also $\alpha'' \cdot \langle b \rangle \cdot \beta \in \mathrm{Tr}_{cc}(p')$, and hence $\alpha' \cdot \langle b \rangle \cdot \beta \in \mathrm{Tr}_{cc}(p)$.
– $\alpha' = \langle b \rangle \cdot \alpha''$. This case is analogous to the previous one. □

A quick reading of Proposition 3 could make us think that the only interesting covariant-contravariant traces that contain some contravariant action in A^l are those which has a single such action, just at the end of the trace. Then, by applying Proposition 3, we would infer that any extension of such a trace is also a trace that we could consider derived from the first, and therefore not so interesting. The wrong implication in this “deduction” comes from the fact that covariant-contravariant traces need not be closed by prefixes, and therefore we could have some trace containing several contravariant actions such that no prefix containing less contravariant actions is also a trace of that process.

Example 2. If $p = ab0$ we have $\langle ab \rangle \notin \mathrm{Tr}_{cc}(p)$ but $\langle abb \rangle \in \mathrm{Tr}_{cc}(p)$.

Next we give a denotational definition of the set $\mathrm{Tr}_{cc}(p)$ by applying the results above.

Proposition 4. *We can alternatively define the set* $\mathrm{Tr}_{cc}(p)$ *as follows:*

– $\mathrm{Tr}_{cc}(0) = \{\langle\rangle\} \cup A^l \cdot A^*$.
– $\mathrm{Tr}_{cc}(a \cdot p) = \{\langle\rangle\} \cup A^l \cdot A^* \cup a \cdot \mathrm{Tr}_{cc}(p)$*, whenever* $a \in A^r$.
– $\mathrm{Tr}_{cc}(b \cdot p) = \{\langle\rangle\} \cup \{\langle b \rangle \cdot \alpha \mid \alpha \in \mathrm{Tr}_{cc}(p) \setminus \{\langle\rangle\}\} \cup \{\langle b' \rangle \cdot \alpha \mid b' \in A^l, b' \neq b, \alpha \in A^*\}$.

– $\mathrm{Tr}_{cc}(p+q) = \{\langle\rangle\} \cup \{\langle a\rangle \cdot \alpha \mid a \in A^r \wedge \langle a\rangle \cdot \alpha \in \mathrm{Tr}_{cc}(p) \cup \mathrm{Tr}_{cc}(q)\} \cup \{\langle b\rangle \cdot \alpha \mid b \in A^l \wedge \langle b\rangle \cdot \alpha \in \mathrm{Tr}_{cc}(p) \cap \mathrm{Tr}_{cc}(q)\}$.

Proof. Let us first prove that Definition 7 implies Proposition 4.

– $\mathrm{Tr}_{cc}(0)$. By definition, $\langle\rangle \in \mathrm{Tr}_{cc}(0)$. Since $0 \overset{b}{\not\longrightarrow}$, $\langle b\rangle \in \mathrm{Tr}_{cc}(0)$ for all $b \in A^l$, and then applying the last clause we obtain that $\langle b\rangle \cdot \alpha \in \mathrm{Tr}_{cc}(0)$ for all $b \in A^l$ and all $\alpha \in A^*$.
– $\mathrm{Tr}_{cc}(a \cdot p)$. By the definition $\mathrm{Tr}_{cc}(a \cdot p) \supseteq \{\langle\rangle\} \cup a \cdot \mathrm{Tr}_{cc}(p)$ but, since $a \cdot p \overset{b}{\not\longrightarrow}$, we must add all the no-traces, that is, $\mathrm{Tr}_{cc}(a \cdot p) = \{\langle\rangle\} \cup A^l \cdot A^* \cup a \cdot \mathrm{Tr}_{cc}(p)$.
– $\mathrm{Tr}_{cc}(b \cdot p)$. $\langle\rangle \in \mathrm{Tr}_{cc}(b \cdot p)$ and, since $b \cdot p \overset{b'}{\not\longrightarrow}$ for any $b' \in A^l$ with $b' \neq b$, $\langle b'\rangle \cdot \alpha \in \mathrm{Tr}_{cc}(b \cdot p)$ for all $b' \neq b$ and $\alpha \in A^*$ by clause 5. Also, since $b \cdot p \overset{b}{\longrightarrow} p$ we have that $\langle b\rangle \cdot \alpha \in \mathrm{Tr}_{cc}(b \cdot p)$ iff $\alpha \neq \langle\rangle$ and $\alpha \in \mathrm{Tr}_{cc}(p)$, as we needed to prove.
– $\mathrm{Tr}_{cc}(p+q)$. Again $\langle\rangle \in \mathrm{Tr}_{cc}(p+q)$. Also, $\langle a\rangle \cdot \alpha \in \mathrm{Tr}_{cc}(p+q)$ if and only if $p \overset{a}{\longrightarrow} p'$ and $\alpha \in \mathrm{Tr}_{cc}(p')$ or $q \overset{a}{\longrightarrow} q'$ and $\alpha \in \mathrm{Tr}_{cc}(q')$, that is, $\{\langle a\rangle \cdot \alpha \mid a \in A^r \wedge \langle a\rangle \cdot \alpha \in \mathrm{Tr}_{cc}(p) \cup \mathrm{Tr}_{cc}(q)\} \subseteq \mathrm{Tr}_{cc}(p+q)$. Analogously, $\langle b\rangle \cdot \alpha \in \mathrm{Tr}_{cc}(p+q)$ if and only if both $p \overset{b}{\not\longrightarrow}$ and $q \overset{b}{\not\longrightarrow}$, or $\alpha \in \mathrm{Tr}_{cc}(p')$ for all $p \overset{b}{\longrightarrow} p'$ and $\alpha \in \mathrm{Tr}_{cc}(q')$ for all $q \overset{b}{\longrightarrow} q'$, that is, $\{\langle b\rangle \cdot \alpha \mid b \in A^l \wedge \langle b\rangle \cdot \alpha \in \mathrm{Tr}_{cc}(p) \cap \mathrm{Tr}_{cc}(q)\} \subseteq \mathrm{Tr}_{cc}(p+q)$.

On the other hand, let us show that Proposition 4 implies Definition 7. For that, let p be a process and consider all possibilities for α to be a covariant-contravariant trace of p.

– $\langle\rangle \in \mathrm{Tr}_{cc}(p)$ for all p. Trivial.
– $\alpha = \langle a\rangle \cdot \alpha'$, $a \in A^r$. Let us prove that if $\alpha \in \mathrm{Tr}_{cc}(p)$ then $p \overset{a}{\longrightarrow} p'$ and $\alpha' \in \mathrm{Tr}_{cc}(p')$. The proof will follow by structural induction over the shape of p.
  - $\langle a\rangle \cdot \alpha' \notin \mathrm{Tr}_{cc}(0)$.
  - Let $\langle a\rangle \cdot \alpha' \in \mathrm{Tr}_{cc}(c \cdot p)$. First, the only way of obtaining a trace beginning with $\langle a\rangle$ is having $c = a$ so let us consider the process $a \cdot p$. Now, by definition $\mathrm{Tr}_{cc}(a \cdot p) = \{\langle\rangle\} \cup A^l \cdot A^* \cup a \cdot \mathrm{Tr}_{cc}(p)$, thus, $a \cdot p \overset{a}{\longrightarrow} p$ and $\alpha' \in \mathrm{Tr}_{cc}(p)$.
  - Let $\langle a\rangle \cdot \alpha' \in \mathrm{Tr}_{cc}(p+q)$. By definition, $\mathrm{Tr}_{cc}(p+q) = \{\langle\rangle\} \cup \{\langle a\rangle \cdot \alpha \mid a \in A^r \wedge \langle a\rangle \cdot \alpha \in \mathrm{Tr}_{cc}(p) \cup \mathrm{Tr}_{cc}(q)\} \cup \{\langle b\rangle \cdot \alpha \mid b \in A^l \wedge \langle b\rangle \cdot \alpha \in \mathrm{Tr}_{cc}(p) \cap \mathrm{Tr}_{cc}(q)\}$, so obtaining the trace $\langle a\rangle \cdot \alpha'$ is possible only if $\langle a\rangle \cdot \alpha' \in \mathrm{Tr}_{cc}(p)$ or $\langle b\rangle \cdot \alpha' \in \mathrm{Tr}_{cc}(q)$ and hence, by induction hypothesis, we obtain $p+q \overset{a}{\longrightarrow} p'$ with $\alpha' \in \mathrm{Tr}_{cc}(p')$ or $p+q \overset{a}{\longrightarrow} q'$ with $\alpha' \in \mathrm{Tr}_{cc}(q')$, as we needed to prove.
– $\alpha = \langle b\rangle$ with $b \in A^l$. Let us prove that if $\alpha \in \mathrm{Tr}_{cc}(p)$ then $p \overset{b}{\not\longrightarrow}$. The proof will follow by structural induction over the shape of p.
  - By definition $\langle b\rangle \in \mathrm{Tr}_{cc}(0)$, and obviously $0 \overset{b}{\not\longrightarrow}$.
  - By definition $\langle b\rangle \in \mathrm{Tr}_{cc}(a \cdot p)$, with $a \in A^r$, and obviously $a \cdot p \overset{b}{\not\longrightarrow}$.
  - Let $\langle b\rangle \in \mathrm{Tr}_{cc}(b' \cdot p')$. Since by definition $\mathrm{Tr}_{cc}(b \cdot p) = \{\langle\rangle\} \cup \{\langle b\rangle \cdot \alpha \mid \alpha \in \mathrm{Tr}_{cc}(p) \setminus \{\langle\rangle\}\} \cup \{\langle b'\rangle \cdot \alpha \mid b' \in A^l, b' \neq b, \alpha \in A^*\}$, the only way of obtaining the trace $\langle b\rangle$ is with the last condition (since in the second one $\langle b\rangle \cdot \alpha$ needs $\alpha \neq \langle\rangle$), that is, if the process p is $b' \cdot p'$ such that $b' \neq b$. Hence, $p \overset{b}{\not\longrightarrow}$.

- Let $\langle b\rangle \in \mathrm{Tr}_{cc}(q+r)$. By definition, the only way of obtaining the trace $\langle b\rangle$ is having $\langle b\rangle \in \mathrm{Tr}_{cc}(q)$ and $\langle b\rangle \in \mathrm{Tr}_{cc}(r)$ and hence, by induction hypothesis, $q+r \stackrel{b}{\not\longrightarrow}$ as we wanted to prove.

– $\alpha = \langle b\rangle \cdot \alpha'$ with $b \in A^l$, $\alpha' \neq \langle\rangle$. As in the previous case let us prove that if $\alpha \in \mathrm{Tr}_{cc}(p)$ then $\alpha' \in \mathrm{Tr}_{cc}(p')$ for all $p \xrightarrow{b} p'$. The proof will follow by structural induction over the shape of p.

- By definition $\langle b\rangle \cdot \alpha' \in \mathrm{Tr}_{cc}(0)$, and obviously $\alpha' \in \mathrm{Tr}_{cc}(p')$ for all $0 \xrightarrow{b} p'$, because there is no such p'.
- By definition $\langle b\rangle\cdot\alpha' \in \mathrm{Tr}_{cc}(a\cdot p)$, and obviously $\alpha' \in \mathrm{Tr}_{cc}(p')$ for all $a\cdot p \xrightarrow{b} p'$, because there is no such p'.
- Let $\langle b\rangle \cdot \alpha' \in \mathrm{Tr}_{cc}(c\cdot q)$. By definition, $\mathrm{Tr}_{cc}(b\cdot p) = \{\langle\rangle\} \cup \{\langle b\rangle \cdot \alpha \mid \alpha \in \mathrm{Tr}_{cc}(p) \setminus \{\langle\rangle\}\} \cup \{\langle b'\rangle \cdot \alpha \mid b' \in A^l, b' \neq b, \alpha \in A^*\}$. Since $\alpha' \neq \langle\rangle$, there are two ways of obtaining the trace α. If $c \neq b$, $c \in A^l$ we have $c\cdot q \stackrel{b}{\not\longrightarrow}$ and thus $\alpha' \in \mathrm{Tr}_{cc}(q')$ for all $c\cdot q \xrightarrow{b} q$. On the other hand, if the process is $p = b\cdot q$, then by definition of its trace, $\alpha' \in \mathrm{Tr}_{cc}(q)$ for $b\cdot q \xrightarrow{b} q$.
- Let $\langle b\rangle \cdot \alpha' \in \mathrm{Tr}_{cc}(q+r)$. By definition, $\mathrm{Tr}_{cc}(q+r) = \{\langle\rangle\} \cup \{\langle a\rangle \cdot \alpha \mid a \in A^r \wedge \langle a\rangle\cdot\alpha \in \mathrm{Tr}_{cc}(q) \cup \mathrm{Tr}_{cc}(r)\} \cup \{\langle b\rangle\cdot\alpha \mid b \in A^l \wedge \langle b\rangle\cdot\alpha \in \mathrm{Tr}_{cc}(q) \cap \mathrm{Tr}_{cc}(r)\}$, that is, the trace $\langle b\rangle \cdot \alpha'$ can only be obtained if $\langle b\rangle\cdot\alpha' \in \mathrm{Tr}_{cc}(q)$ and $\langle b\rangle\cdot\alpha' \in \mathrm{Tr}_{cc}(r)$ and hence, by induction hypothesis, we obtain the result. □

Example 3. Given $p = a$, $q = ab + ab'$, $r = a(b + b')$, where $a \in A^r$ and $b, b' \in A^l$, with $b' \neq b$, we obtain that $p =_{ccT} q$ but $q \neq_{ccT} r$. Indeed, $\mathrm{Tr}_{cc}(p) = \mathrm{Tr}_{cc}(q) = \{\langle\rangle, \langle a\rangle\} \cup A^l \cdot A^* \cup \langle a\rangle \cdot A^l \cdot A^*$; note that, for example, $\langle a, b\rangle \in \mathrm{Tr}_{cc}(q)$ since we have $a \cdot b' \cdot 0 \xrightarrow{a} b' \cdot 0 \stackrel{b}{\not\longrightarrow}$. However, $\langle ab\rangle \notin \mathrm{Tr}_{cc}(r)$, so that $r \lesssim_{ccT} p$, but $p \not\lesssim_{ccT} r$.

But note that given the processes $p = aba' + aba''$, $q = a(ba' + ba'')$ and $r = ab(a'+a'')$ we have $p =_{ccT} r$ and $p \neq_{ccT} q$. This is because the trace $\langle aba'\rangle \in \mathrm{Tr}_{cc}(p)$, but $a(ba'+ba'') \xrightarrow{a} ba' + ba''$ and by Proposition 4 the traces begining with an action in A^l are as follows: $\mathrm{Tr}_{cc}(ba'+ba'') = \{\langle b\rangle\cdot\alpha \mid \langle b\rangle\cdot\alpha \in \mathrm{Tr}_{cc}(ba') \cap \mathrm{Tr}_{cc}(ba'')\}$. But, since $\langle b\rangle \cdot \alpha \notin \mathrm{Tr}_{cc}(ba')$ implies $\alpha = \langle a'\rangle$ and $\langle b\rangle \cdot \alpha \notin \mathrm{Tr}_{cc}(ba'')$ implies $\alpha = \langle a''\rangle$ there is no trace starting with b in $\langle ba' + ba''\rangle$, and thus $\langle aba'\rangle \notin \mathrm{Tr}_{cc}(q)$ (observe also that neither $\langle b\rangle \in \mathrm{Tr}_{cc}(ba')$ nor $\langle b\rangle \in \mathrm{Tr}_{cc}(ba'')$, since $ba' \xrightarrow{b}$ and $ba'' \xrightarrow{b}$).

As one would expect, the covariant-contravariant simulation order $\lesssim_{cc}$ is finer than the covariant-contravariant trace preorder $\lesssim_{ccT}$.

Proposition 5. *For all processes p, q over the set of actions $A = A^r \cup A^l$, we have: if $p \lesssim_{cc} q$, then $p \lesssim_{ccT} q$.*

Proof. We check that $p \lesssim_{cc} q$ and $\alpha \in \mathrm{Tr}_{cc}(p)$ implies that $\alpha \in \mathrm{Tr}_{cc}(q)$, by structural induction over the form of a trace $\alpha \in A^*$.

– $\alpha = \langle\rangle$: trivial.

– $\alpha = \langle b\rangle$, with $b \in A^l$. Since $p \lesssim_{cc} q$, if $p \stackrel{b}{\not\longrightarrow}$ then $q \stackrel{b}{\not\longrightarrow}$ and $\langle b\rangle \in \mathrm{Tr}_{cc}(q)$.

– $\alpha = \langle a \rangle \cdot \alpha'$, with $a \in A^r$. By definition this means that there exists p' such that $p \xrightarrow{a} p'$ and $\alpha' \in \mathrm{Tr}_{cc}(p')$. Since $p \lesssim_{cc} q$ and $p \xrightarrow{a} p'$, there exists q' such that $q \xrightarrow{a} q'$ with $p' \lesssim_{cc} q'$. So, by induction hypothesis $\mathrm{Tr}_{cc}(p') \subseteq \mathrm{Tr}_{cc}(q')$, hence $\alpha' \in \mathrm{Tr}_{cc}(q')$ and $\alpha \in \mathrm{Tr}_{cc}(q)$.
– $\alpha = \langle b \rangle \cdot \alpha'$, with $b \in A^l$, and $\alpha' \neq \langle\rangle$. By definition either $p \stackrel{b}{\not\longrightarrow}$ or $\alpha' \in \mathrm{Tr}_{cc}(p')$ for every $p \xrightarrow{b} p'$. If $p \stackrel{b}{\not\longrightarrow}$, since $p \lesssim_{cc} q$, we also have that $q \stackrel{b}{\not\longrightarrow}$ and thus $\langle b \rangle \cdot \alpha' \in \mathrm{Tr}_{cc}(q)$. On the other hand, if we have that $p \xrightarrow{b} p'$, then also $\alpha' \in \mathrm{Tr}_{cc}(p')$. We have to show that $q \xrightarrow{b} q'$ implies $\alpha' \in \mathrm{Tr}_{cc}(q')$ but, since $p \lesssim_{cc} q$, for all q' such that $q \xrightarrow{b} q'$ we have some $p \xrightarrow{b} p'$ and $p' \lesssim_{cc} q'$, and then applying the induction hypothesis we obtain that $\alpha' \in \mathrm{Tr}_{cc}(q')$, and therefore $\alpha \in \mathrm{Tr}_{cc}(q)$. □

Example 4. Let $p = a\cdot(b\cdot 0 + a\cdot 0)$ and $q = a\cdot a\cdot 0$. It is easy to check that $p \lesssim_{cc} q$ thus, by Proposition 5 we have that $\mathrm{Tr}_{cc}(p) \subseteq \mathrm{Tr}_{cc}(q)$. For example, we have $\langle abb \rangle \in \mathrm{Tr}_{cc}(p)$ and since $q \xrightarrow{a} a \cdot 0$ and $a \cdot 0 \stackrel{b}{\not\longrightarrow}$ we also have that $\langle abb \rangle \in \mathrm{Tr}_{cc}(q)$. On the other hand, we also have that $\langle ab \rangle \in \mathrm{Tr}_{cc}(q)$, but $\langle ab \rangle \notin \mathrm{Tr}_{cc}(p)$ since $p \xrightarrow{a} (b \cdot 0 + a \cdot 0)$ and $(b \cdot 0 + a \cdot 0) \xrightarrow{b} 0$.

4 Trace semantics for conformance simulation

Next we develop the adequate notion of conformance traces which defines a reasonable trace preorder that fits well under the conformance simulation preorder developed in [6]. Conformance traces are those that not only can be executed, but cannot be refused (from the root of the process) at any time during its execution: therefore we could also call them "secure traces".

Definition 8. *We say that $\alpha = a_1 \ldots a_n$ is a conformance trace of the process p (denoted by $\alpha \in \mathrm{Tr}_{cf}(p)$) if and only if*

– $\alpha = \langle\rangle$, *or*
– $\alpha = \langle a \rangle$ *and* $p \xrightarrow{a}$, *or*
– $\alpha = \langle a \rangle \cdot \alpha'$, $p \xrightarrow{a}$ *and* $\alpha' \in \mathrm{Tr}_{cf}(p')$ *for all* $p \xrightarrow{a} p'$.

We define the conformance trace preorder $\lesssim_{cfT}$ by $p \lesssim_{cfT} q$ iff $\mathrm{Tr}_{cf}(p) \subseteq \mathrm{Tr}_{cf}(q)$.

Once again, the second clause is redundant and $\mathrm{Tr}_{cf}(p)$ is the smallest set of sequences that satisfies the conditions above. For understanding better why we call conformance traces secure traces, let us state and prove the following lemma.

Lemma 1. *If $\alpha \cdot \beta$ is a conformance trace of a process p, then $p' \xrightarrow{\beta}$ for any $p \xrightarrow{\alpha} p'$.*

Proof. Let us suppose that there exists some p'_0 such that $p \xrightarrow{\alpha} p'_0$ and $p'_0 \stackrel{\beta}{\not\longrightarrow}$. But, since $\alpha \cdot \beta \in \mathrm{Tr}_{cf}(p)$, by definition $\beta \in \mathrm{Tr}_{cf}(p'_0)$ and, in particular, this means that $p'_0 \xrightarrow{\beta}$. □

Example 5. For example, for the deterministic processes $p = a \cdot b \cdot 0$ and $q = a \cdot a \cdot 0$, we have $\mathrm{Tr}_{cf}(p) = \mathrm{Tr}(p) = \{\langle\rangle, \langle a\rangle, \langle ab\rangle\}$ and $\mathrm{Tr}_{cf}(q) = \mathrm{Tr}(q) = \{\langle\rangle, \langle a\rangle, \langle aa\rangle\}$.

Now, let us consider the non-deterministic process $r = p + q$, and let us compute the conformance traces of r. Obviously, $\langle\rangle, \langle a\rangle \in \mathrm{Tr}_{cf}(r)$; and now $\langle a\rangle \cdot \alpha' \in \mathrm{Tr}_{cf}(r)$ implies that $\alpha' \in \mathrm{Tr}_{cf}(r')$ for all $r \xrightarrow{a} r'$, in particular, $\alpha' \in \mathrm{Tr}_{cf}(b \cdot 0)$ and $\alpha' \in \mathrm{Tr}_{cf}(a \cdot 0)$, that is, the only possible trace α' is $\alpha' = \langle\rangle$. Hence, $\{\langle\rangle, \langle a\rangle\} = \mathrm{Tr}_{cf}(r)$. It is in this sense that we can also call secure traces to the conformance traces, since from the root of the process r we know that the process is not going to refuse any of their conformance traces, whereas if we would try to execute the trace $\langle a, a\rangle$ it could be refused by r.

From this example we obtain that, in particular, if a process is deterministic its conformance traces coincide with its (plain) traces, as the following lemma makes precise.

Lemma 2. *If p is a deterministic process then $\mathrm{Tr}(p) = \mathrm{Tr}_{cf}(p)$.*

Proof. It suffices to observe that in the case than p is deterministic the universal quantification in the third clause of Definition 8 can be seen as an existential quantification (because for deterministic processes there is only one transition labeled with a concrete action a). □

Let us now show that the conformance simulation preorder is finer than the conformance trace preorder.

Proposition 6. *For any processes p, q, if $p \lesssim_{cf} q$ then $p \lesssim_{cfT} q$.*

Proof. We need to show that $p \lesssim_{cf} q$ and $\alpha \in \mathrm{Tr}_{cf}(p)$ imply $\alpha \in \mathrm{Tr}_{cf}(q)$. We will prove it by structural induction over the shape of a trace α.

- $\alpha = \langle\rangle$: trivial.
- $\alpha = \langle a\rangle \cdot \alpha'$: by definition, $p \xrightarrow{a}$ and if $p \xrightarrow{a} p'$ then $\alpha' \in \mathrm{Tr}_{cf}(p')$. Using that $p \lesssim_{cf} q$ and $p \xrightarrow{a}$, we obtain that $q \xrightarrow{a}$ and for every $q \xrightarrow{a} q'$ there exists $p \xrightarrow{a} p'$ such that $p' \lesssim_{cf} q'$. So, since $\alpha' \in \mathrm{Tr}_{cf}(p')$, applying the induction hypothesis we get $\alpha' \in \mathrm{Tr}_{cf}(q')$, and therefore $\alpha \in \mathrm{Tr}_{cc}(q)$. □

A denotational definition of the set of conformance traces of a process is also possible:

Proposition 7. *We can alternatively define the set $\mathrm{Tr}_{cf}(p)$ as follows:*

- $\mathrm{Tr}_{cf}(0) = \{\langle\rangle\}$.
- $\mathrm{Tr}_{cf}(a \cdot p) = \{\langle\rangle\} \cup \langle a\rangle \cdot \mathrm{Tr}_{cc}(p)$.
- $\mathrm{Tr}_{cf}(p+q) = \mathrm{Tr}_{cf}(p) \Cup \mathrm{Tr}_{cf}(q)$, *where $\Cup$ denotes the following modified definition of union-intersection for sets of traces: $T_1 \Cup T_2 = (T_1 \cap T_2) \cup \{\langle a\rangle \cdot \alpha \in T_1 \cup T_2 \mid \langle a\rangle \notin T_1 \cap T_2\}$.*

Proof. Let us first prove that this characterization follows from Definition 8.

- $\mathrm{Tr}_{cf}(0) = \langle\rangle$ trivially.

– $\mathrm{Tr}_{cf}(a \cdot p)$. Obviously $\langle\rangle \in \mathrm{Tr}_{cf}(a \cdot p)$. Also, since $a \cdot p \xrightarrow{a} p$, if $\alpha' \in \mathrm{Tr}_{cf}(p)$ then $\langle a \rangle \cdot \alpha' \in \mathrm{Tr}_{cf}(a \cdot p)$. In other words, $\mathrm{Tr}_{cf}(a \cdot p) = \langle\rangle \cup \langle a \rangle \cdot \mathrm{Tr}_{cf}(p)$.
– $\mathrm{Tr}_{cf}(p + q)$. Obviously $\langle\rangle \in \mathrm{Tr}_{cf}(p + q)$, $\langle\rangle \in \mathrm{Tr}_{cf}(p)$ and $\langle\rangle \in \mathrm{Tr}_{cf}(q)$. Now, for the actions $a \in I(p) \cap I(q)$, $\langle a \rangle \cdot \alpha \in \mathrm{Tr}_{cf}(p + q)$ implies $\langle a \rangle \cdot \alpha \in \mathrm{Tr}_{cf}(p)$ and $\langle a \rangle \cdot \alpha \in \mathrm{Tr}_{cf}(q)$. For the actions $a \notin I(p) \cap I(q)$, that is, $a \in I(p) \setminus I(q)$ (respectively $a \in I(q) \setminus I(p)$) if $\langle a \rangle \cdot \alpha \in \mathrm{Tr}_{cf}(p)$ (respectively $\langle a \rangle \cdot \alpha \in \mathrm{Tr}_{cf}(q)$) then $\langle a \rangle \cdot \alpha \in \mathrm{Tr}_{cf}(p+q)$. Thus, we have proved $\mathrm{Tr}_{cf}(p+q) = \mathrm{Tr}_{cf}(p) \uplus \mathrm{Tr}_{cf}(q)$.

Analogously, let us prove that the conditions in Proposition 7 imply Definition 8.

– $\alpha = \langle\rangle$ is trivially a conformance trace for all processes.
– $\alpha = \langle a \rangle \cdot \alpha'$. Let us prove that if $\alpha \in \mathrm{Tr}_{cc}(p)$ then $p \xrightarrow{a}$ and $\alpha' \in \mathrm{Tr}_{cc}(p')$ for all $p \xrightarrow{a} p'$. The proof will follow by structural induction over the shape of p.
  - Let $p = a \cdot q$. By the characterization, $\mathrm{Tr}_{cf}(a \cdot p) = \langle\rangle \cup \langle a \rangle \cdot \mathrm{Tr}_{cf}(q)$, so $p \xrightarrow{a} q$ and $\alpha \in \mathrm{Tr}_{cf}(q)$.
  - Let $p = q + r$. By the characterization, $\mathrm{Tr}_{cf}(q + r) = \mathrm{Tr}_{cf}(q) \uplus \mathrm{Tr}_{cf}(r)$, so if $a \in I(q) \cap I(r)$ then $\alpha \in \mathrm{Tr}_{cc}(q)$ and $\alpha \in \mathrm{Tr}_{cc}(r)$ and, by induction hypothesis, $\alpha' \in \mathrm{Tr}_{cc}(p')$ for all $p \xrightarrow{a} p'$. On the other hand, if $a \in I(q) \cup I(r)$ but $a \notin I(q) \cap I(r)$ then $\alpha \in \mathrm{Tr}_{cc}(q)$ or $\alpha \in \mathrm{Tr}_{cc}(r)$: in either case, again by induction hypothesis we obtain the result.

□

Unfortunately, this semantics has an important drawback: the conformance trace preorder is not a precongruence for the operators in *BCCSP(A)*.

Example 6. Obviously we have both $0 \lesssim_{cfT} ab$ and $ac \lesssim_{cfT} ac$, but $ac \not\lesssim_{cfT} ac + ab$. This problem already appeared in the conformance simulation preorder and it does not disappear even though the conformance trace preorder is coarser.

It is easy to see that the problem can be solved exactly as it was done for the conformance simulation preorder in [7].

Definition 9. *We define the observational conformance trace preorder $\lesssim_{cfT}^p$ as:*

$$p \lesssim_{cfT}^p q \iff I(p) = I(q) \wedge p \lesssim_{cfT} q \; .$$

Proposition 8. *$\lesssim_{cfT}^p$ is the biggest precongruence contained in $\lesssim_{cfT}$.*

Proof. Obviously, we have $\lesssim_{cfT}^p \subseteq \lesssim_{cfT}$. If there were a larger precongruence, it would contain p and q with $p \lesssim_{cfT} q$ but $I(q) \neq I(p)$: then, taking $a \in I(q) \setminus I(p)$ and, assuming A big enough, taking $b \in A$ such that $q \stackrel{a \cdot b}{\not\longrightarrow}$ we would have $ab+p \not\lesssim_{cfT} ab+q$ (since $ab \not\lesssim_{cfT} q$).

Finally, both the prefix operator and $+$ preserve $\lesssim_{cfT}^p$:

– If $p \lesssim_{cfT}^p q$, then $ap \lesssim_{cfT}^p aq$ since $I(ap) = I(aq) = \{a\}$, and $\langle a \rangle \cdot \alpha' \in \mathrm{Tr}_{cf}(ap)$ with $\alpha' \in \mathrm{Tr}_{cf}(p)$, hence, $ap \lesssim_{cfT} aq$.

– If $p \lesssim^p_{cfT} q$, we must show that also $p + r \lesssim^p_{cfT} q + r$. First, we have $I(p + r) = I(q + r) = I(r) \cup I(p)$; and by definition $\mathrm{Tr}_{cf}(p + r) = \mathrm{Tr}_{cf}(p) \Cup \mathrm{Tr}_{cf}(r) = (\mathrm{Tr}_{cf}(p) \cap \mathrm{Tr}_{cf}(r)) \cup \{\langle a\rangle \cdot \alpha \in I(p) \cup I(r) \mid \langle a\rangle \notin I(p) \cap I(r)\} = (\mathrm{Tr}_{cf}(p) \cap \mathrm{Tr}_{cf}(r)) \cup \{\langle a\rangle \cdot \alpha \in I(p) \cup I(r) \mid \langle a\rangle \notin I(q) \cap I(r)\}$, which is included in $(\mathrm{Tr}_{cf}(q) \cap \mathrm{Tr}_{cf}(r)) \cup \{\langle a\rangle \cdot \alpha \in I(q) \cup I(r) \mid \langle a\rangle \notin I(q) \cap I(r)\} = \mathrm{Tr}_{cf}(q) \Cup \mathrm{Tr}_{cf}(r) = \mathrm{Tr}_{cf}(q + r)$ because by hypothesis $\mathrm{Tr}_{cf}(p) \subseteq \mathrm{Tr}_{cf}(q)$. Thus, obtaining $p + r \lesssim^p_{cfT} q + r$. □

5 Covariant-contravariant ready simulation and its induced linear semantics

Following the unified classification of the semantics for (covariant) processes in [5], the most well-known linear semantics (failures, readiness, ...) are related not with plain simulation semantics, as is the case for the trace semantics, but with the ready simulation semantics. Therefore, before generalizing the definition of those linear semantics it is natural to consider the case of ready simulation.

Covariant-contravariant ready simulations are obtained exactly as in the plain case by simply constraining the definition of covariant-contravariant simulation by imposing the condition (I) $I(p) = I(q)$ to all the pairs of related processes p, q.

Definition 10. *Given two processes p and q, and $\{A^r, A^l\}$ a partition of the alphabet A, a covariant-contravariant ready simulation between them is a relation R such that for every pRq we have:*

- *For all $a \in A^r$ and all $p \xrightarrow{a} p'$ there exists $q \xrightarrow{a} q'$ with $p'Rq'$.*
- *For all $a \in A^l$, and all $q \xrightarrow{a} q'$ there exists $p \xrightarrow{a} p'$ with $p'Rq'$.*
- *$I(p) = I(q)$.*

We will denote the induced preorder by $\lesssim_{Rcc}$.

Trivially, we have that the covariant-contravariant ready simulation preorder is finer than the covariant-contravariant simulation preorder.

5.1 Covariant-contravariant failures semantics

Once again, in order to motivate our definitions we start by considering the case of the pure contravariant failure semantics which, of course, is obtained by dualizing the definition of (covariant) failures.

Definition 11 (Contravariant failures semantics). *We say that $\langle\beta, Y\rangle$ is a contravariant failure pair of a process p if $I(p') \cap Y \neq \emptyset$ for all processes p' such that $p \xrightarrow{\beta} p'$. We denote the set of contravariant failures of p by $\mathrm{F}_c(p)$, and by $\lesssim_{cF}$ the order induced by comparing the set of contravariant failures of two contravariant processes.*

Proposition 9. $\mathrm{F}_c(p) = \overline{\mathrm{F}(p)}$, *for all processes p*[3].

[3] As before, we assume that the (covariant) actions in p become contravariant ones to obtain this result.

Proof. $\langle\beta, Y\rangle \in \mathrm{F}_c(p)$ if and only if $I(p') \cap Y \neq \emptyset$ for all $p \xrightarrow{\beta} p'$. In other words, if there is no process p'' such that $p \xrightarrow{\beta} p''$ with $I(p'') \cap Y = \emptyset$, that is, $\langle\beta, Y\rangle \notin \mathrm{F}(p)$. □

In order to define the most reasonable notion of covariant-contravariant failures semantics, we keep separated the covariant and contravariant failures but considering now any arbitrary "trace"[4] to produce along a covariant-contravariant process.

Definition 12 (Extended covariant failures). *Given $\gamma \in A^*$ and $R \subseteq A^r$, we say that $\langle\gamma, R\rangle$ is an extended covariant failure pair of a covariant-contravariant process p, written $\langle\gamma, R\rangle \in \mathrm{F}(p)$, if*

- $\gamma = \langle\rangle$, $I(p) \cap R = \emptyset$*; or*
- $\gamma = \langle a\rangle \cdot \gamma'$ *with* $a \in A^r$*, and there exists* $p \xrightarrow{a} p'$ *such that* $\langle\gamma', R\rangle \in \mathrm{F}(p')$*; or*
- $\gamma = \langle b\rangle \cdot \gamma'$ *with* $b \in A^l$*, and* $\langle\gamma', R\rangle \in \mathrm{F}(p')$ *for all* $p \xrightarrow{b} p'$.

By abuse of notation, we will also denote by $\lesssim_F$ the induced order over covariant-contravariant processes, since it can be easily checked that it extends the corresponding order over covariant processes.

Definition 13 (Extended contravariant failures). *Given $\gamma \in A^*$ and $L \subseteq A^l$, we say that $\langle\gamma, L\rangle$ is an extended contravariant failure pair of a covariant-contravariant process p, written $\langle\gamma, L\rangle \in \mathrm{F}_c(p)$, if*

- $\gamma = \langle\rangle$, $I(p) \cap L \neq \emptyset$*; or*
- $\gamma = \langle a\rangle \cdot \gamma'$ *with* $a \in A^r$*, and there exists* $p \xrightarrow{a} p'$ *such that* $\langle\gamma', L\rangle \in \mathrm{F}_c(p')$*; or*
- $\gamma = \langle b\rangle \cdot \gamma'$ *with* $b \in A^l$*, and* $\langle\gamma', L\rangle \in \mathrm{F}_c(p')$ *for all* $p \xrightarrow{b} p'$.

We will also denote by $\lesssim_{cF}$ the obtained preorder over covariant-contravariant processes, since it generalizes the corresponding notion over contravariant processes.

Note that in the two definitions above we continue applying the "alternating" ∃,∀ procedure along the sequence γ, depending on the covariant or contravariant character of the executed actions.

Hence, we define the covariant-contravariant failures preorder as the conjunction of the two previously defined orders.

Definition 14. *We define $p \lesssim_{ccF} q$ if and only if $p \lesssim_F q$ and $p \lesssim_{cF} q$. That is, if the process q has more extended covariant failures than p and more extended contravariant failures.*

[4] We have here the word "trace" between quotation marks because, as we will show, the sequences γ that appear in our covariant-contravariant failure need not be a covariant-contravariant trace of the corresponding process.

The obtained failures semantics is finer than the covariant-contravariant traces semantics, since from the corresponding set of failures pairs we can recover the set of covariant-contravariant traces of any process. However, it is not always true that any sequence γ is a covariant-contravariant failure pair corresponds to a covariant-contravariant trace of the process. Finally, we will see that the covariant-contravariant failures semantics is indeed coarser than the covariant-contravariant ready simulation semantics.

Lemma 3. *Given $\gamma \in A^*$, we have the following:*

– $\langle \gamma, \emptyset \rangle \in \mathrm{F}_c(p)$ *if and only if* $\gamma \in \mathrm{Tr}_{cc}(p)$ *and p cannot perform all the actions in γ, that is,* $p \stackrel{\gamma}{\not\longrightarrow}$.
– *Given* $\gamma \neq \gamma' \cdot \langle b \rangle$, $\langle \gamma, \emptyset \rangle \in \mathrm{F}(p)$ *if and only if* $\gamma \in \mathrm{Tr}_{cc}(p)$ *and p can perform all the actions in γ, that is,* $p \stackrel{\gamma}{\longrightarrow}$.

Proof. First we observe that there are two types of covariant-contravariant traces: those that the process can perform, and those that have a contravariant action that the process cannot perform. In this second case the trace must end with a covariant action $a \in A^r$.

So, let us suppose that $\langle \gamma, \emptyset \rangle \in \mathrm{F}_c(p)$. If that is the case, we must have $p \stackrel{\gamma}{\not\longrightarrow}$, because if that is not the case, after performing γ, we will be asking for $I(p') \cap \emptyset \neq \emptyset$ for some p', which is false. Also, in that case, by definition of extended contravariant traces, it is not difficult to see that $\gamma \in \mathrm{Tr}_{cc}(p)$. Analogously, if we have $\gamma \in \mathrm{Tr}_{cc}(p)$ and $p \stackrel{\gamma}{\not\longrightarrow}$ we must also have $\langle \gamma, \emptyset \rangle \in \mathrm{F}_c(p)$.

Now, let $\langle \gamma, \emptyset \rangle \in \mathrm{F}(p)$. Note that, by definition of extended covariant failures and covariant-contravariant traces, the only posible situation for which $\gamma \notin \mathrm{Tr}_{cc}(p)$ is when $\gamma = \gamma' \cdot \langle b \rangle$ and $p \stackrel{\gamma}{\longrightarrow}\stackrel{b}{\longrightarrow}$. So when considering $\gamma \neq \gamma' \cdot \langle b \rangle$ we obtain the second clause of the lemma. □

Proposition 10. *If $p \lesssim_{ccF} q$ then, $p \lesssim_{ccT} q$.*

Proof. Immediate from Lemma 3. □

Proposition 11. *Given processes p, q, we have $p \lesssim_{Rcc} q$ implies $p \lesssim_{ccF} q$.*

Proof. We check that $p \lesssim_{Rcc} q$ implies $p \lesssim_F q$ and $p \lesssim_{cF} q$. So, let $\langle \gamma, R \rangle \in \mathrm{F}(p)$ and $\langle \rho, L \rangle \in \mathrm{F}_c(p)$, and let us prove that $\langle \gamma, R) \rangle \in \mathrm{F}(q)$ and $\langle \rho, L) \rangle \in \mathrm{F}_c(q)$, by structural induction over the shape of the traces $\gamma \in A^*$ and $\rho \in A^*$.

– $\gamma = \langle \rangle$. By definition $I(p) \cap R = \emptyset$ and since $p \lesssim_{Rcc} q$ implies $I(p) = I(q)$, then $\langle \gamma, R \rangle \in \mathrm{F}(q)$. The case $\rho = \langle \rangle$ is analogous.
– $\gamma = \langle a \rangle \cdot \gamma'$, with $a \in A^r$. By definition this means that there exists p' such that $p \stackrel{a}{\longrightarrow} p'$ and $\langle \gamma', R \rangle \in \mathrm{F}(p')$. Since $p \lesssim_{Rcc} q$ and $p \stackrel{a}{\longrightarrow} p'$, there exists q' such that $q \stackrel{a}{\longrightarrow} q'$ with $p' \lesssim_{Rcc} q'$. So, by induction hypothesis, $\mathrm{F}(p') \subseteq \mathrm{F}(q')$, hence $\langle \gamma', R \rangle \in \mathrm{F}(q')$ and thus, $\langle \gamma, R \rangle \in \mathrm{F}(q)$. Analogous for $\rho = \langle a \rangle \cdot \rho'$

– $\gamma = \langle b\rangle \cdot \gamma'$, with $b \in A^l$. By definition, if $p \xrightarrow{b} p'$ then also $\langle\gamma', R\rangle \in \mathrm{F}(p')$. We have to show that $q \xrightarrow{b} q'$ implies $\langle\gamma', R\rangle \in \mathrm{F}(q')$ but, since $p \lesssim_{Rcc} q$, for all q' such that $q \xrightarrow{b} q'$ we have some $p \xrightarrow{b} p'$ and $p' \lesssim_{Rcc} q'$, and then applying the induction hypothesis we obtain that $\langle\gamma', R\rangle \in \mathrm{F}(q')$, and therefore $\langle\gamma, R\rangle \in \mathrm{F}(q)$. The case $\rho = \langle b\rangle \cdot \rho'$ is analogous. □

5.2 Some examples

By means of some simple examples we are going to illustrate here how the different semantics distinguish less or more pairs of some processes.

Example 7. Let $p = b0$, $q = ba0 + b0$ and $r = bb0 + b0$. We have the following relations:

– $p =_{ccT} q$. Taking $\gamma = \langle b\rangle \cdot \gamma'$ with $\gamma' \neq \langle a\rangle$ it is clear that $\gamma \in \mathrm{Tr}_{cc}(p)$ iff $\gamma \in \mathrm{Tr}_{cc}(q)$, but for $\gamma = \langle ba\rangle$ we have $\gamma \notin \mathrm{Tr}_{cc}(p)$ and $\gamma \notin \mathrm{Tr}_{cc}(q)$, because in order to have $\gamma \in \mathrm{Tr}_{cc}(ba0 + b0)$ we need both $\gamma \in \mathrm{Tr}_{cc}(ba0)$ and $\gamma \in \mathrm{Tr}_{cc}(b0)$. Note also that $b0 \lesssim_{ccT} ba0$, but $ba0 \not\lesssim_{ccT} b0$.
  Instead, we have $r \lesssim_{ccT} p$ but $p \not\lesssim_{ccT} r$ because $\langle bb\rangle \in \mathrm{Tr}_{cc}(p)$ but $\langle bb\rangle \notin \mathrm{Tr}_{cc}(r)$. In this case the fact that $bb0$ can do the second b make this process worse than $b0$, and then r also becomes worse than it.
– $q \lesssim_{ccF} p$ and $p \not\lesssim_{ccF} q$ since, for example, $\langle\langle b\rangle, \{a\}\rangle \in \mathrm{F}(p)$ and $\langle\langle b\rangle, \{a\}\rangle \notin \mathrm{F}(q)$. But covariant-contravariant failures can "see" that, in fact, $b0$ fails more (with the extended notion of covariant failures) than a process that also offers an a as q. Note that we have instead $p =_{cF} q$.
  Also $r \lesssim_{ccF} p$ and $r \lesssim_{ccF} q$, but neither $p \not\lesssim_{ccF} r$ nor $q \not\lesssim_{ccF} r$, because $\langle\langle bb\rangle, \emptyset\rangle \in \mathrm{F}_c(p)$ and $\langle\langle bb\rangle, \emptyset\rangle \in \mathrm{F}_c(q)$, but $\langle\langle bb\rangle, \emptyset\rangle \notin \mathrm{F}_c(r)$ just because $bb0$ can perform the second b.
– $q \lesssim_{cc} p$ and $p \not\lesssim_{cc} q$. Obviously since p has less contravariant actions. Also, $r \lesssim_{cc} q$ and $q \not\lesssim_{cc} r$.
– The three processes are not related with covariant-contravariant ready simulation, that is, $p \not\lesssim_{Rcc} q$, $q \not\lesssim_{Rcc} p$, $q \not\lesssim_{Rcc} r$, $r \not\lesssim_{Rcc} q$, $r \not\lesssim_{Rcc} p$ and $p \not\lesssim_{Rcc} r$.

Example 8. Let us consider the following three machines, where, as usual, the actions whose name terminate with ? are inputs (elements of A^r) and the actions whose name terminate with ! are outputs (elements of A^l).

$$\begin{array}{ll} m_1 & : \ \mathsf{coin?}\ (\mathsf{coke!}\ \mathsf{btt_1?}\ \mathsf{freefanta!}\ 0 + \mathsf{coke!}\ \mathsf{btt_2?}\ \mathsf{free7up!}\ 0) \\ m_2 & : \ \mathsf{coin?}\ (\mathsf{coke!}\ \mathsf{btt_1?}\ \mathsf{freeFanta!}\ 0 + \mathsf{coke!}\ \mathsf{btt_2?}\ \mathsf{free7up!}\ 0 \\ & \qquad + \mathsf{coke!}\ (\mathsf{btt_2?}\ \mathsf{free7up!}\ 0 + \mathsf{btt_1?}\ \mathsf{freeFanta!}\ 0) \\ m_3 & : \ \mathsf{coin?}\ \mathsf{coke!}\ 0 \end{array}$$

m_1 is a machine that after accepting a coin, it chooses between giving us a coke and then allows us to ask for a free fanta, or giving us a coke and then allows us to ask for a free 7-up. The machine m_2 behaves like m_1, but with an extra option in which we can choose either the fanta or the 7-up. Finally, m_3 just gives us a coke for a coin. Now,

– $m_1 =_{ccT} m_3$ and $m_2 =_{ccT} m_3$. Like in the previous example, since when we have a contravariant action (in this case $\mathsf{coke!}$) we are taking the intersection of the traces of its subprocesses, we have that $\langle \mathsf{coin?}\ \mathsf{coke!}\ \mathsf{btt}_1 \rangle \notin \mathrm{Tr}_{cc}(m_i)$, $\langle \mathsf{coin?}\ \mathsf{coke!}\ \mathsf{btt}_2 \rangle \notin \mathrm{Tr}_{cc}(m_i)$ with $i \in \{1,2,3\}$.
– $m_1 =_{ccF} m_2$, because extended covariant failures cannot distinguish if we are in a state that offers $\{\mathsf{btt}_1, \mathsf{btt}_2\}$, $\{\mathsf{btt}_1\}$ or $\{\mathsf{btt}_2\}$. Of course, $m_1 \lesssim_{ccF} m_3$, since $\langle \langle \mathsf{coin?}\ \mathsf{coke!} \rangle, \{btt_i\} \rangle \in \mathrm{F}(m_3)$, that is, after giving us the coke, m_3 fails when we try to press the button $\mathsf{btt_i}$ for all i.
– $m_2 \lesssim_{cc} m_1$, $m_1 \not\lesssim_{cc} m_2$ and $m_1 \lesssim_{cc} m_3$.
– $m_1 \not\lesssim_{Rcc} m_2$ and $m_2 \lesssim_{Rcc} m_1$.

Example 9. Let $p = ba0 + b(a0 + a'0)$ and $q = b(a0 + a'0)$. So, since $p \lesssim_{Rcc} q$ we also have $p \lesssim_{cc} q$, $p \lesssim_{ccF} q$ and $p \lesssim_{ccT} q$.

6 Conformance ready simulations and its induced linear semantics

Ready conformance simulation is also defined by imposing to the conformance simulations the condition (I).

Definition 15. *We say that $R \subseteq X \times Y$ is a ready conformance simulation if for each pair pRq qe have*

– *$I(p) = I(q)$, and*
– *for all $a \in A$ such that $q \xrightarrow{a} q'$ and $p \xrightarrow{a}$, there exists some p' with $p \xrightarrow{a} p'$ and $p'Rq'$.*

We will denote the induced preorder by $\lesssim_{Rcf}$.

In [7] we already proved that the ready conformance simulation preorder is the inverse of the plain ready simulation preorder.

Proposition 12 ([7]). *For all processes p, q we have $p \lesssim_{Rcf} q$ iff $q \lesssim_{RS} p$.*

As an immediate consequence of this result we conclude that the adequate notions of failures, readiness, failure traces and ready traces for the conformance framework are just the complementary of the corresponding plain linear semantics.

As a consequence, the induced equivalences are the same as those for the plain (covariant) semantics, while the induced orders are just the iverse of those.

Proposition 13 ([7]). *For all processes p, q and $X \in \{\mathrm{F}, \mathrm{R}, \mathrm{FT}, \mathrm{RT}\}$ we have $p \lesssim_{cfX} q \iff q \lesssim_X p$, and therefore $p =_{cfX} q \iff p =_X q$.*

Although this could be considered a question of convention, the "direction" of the orders $\lesssim_{cfX}$ reflect in a more natural way the motto "the less non-deterministic, the best" which corresponds to the usual procedure when systems developing (more) deterministic processes refine non-deterministic specifications.

Next, we present as an example, the definition of conformance failures, which by Proposition 9 is the "opposite" of plain failures.

Definition 16. *We say that $\langle \beta, Y \rangle$ is a conformance failure pair of a process p if $I(p') \cap Y \neq \emptyset$ for all processes p' such that $p \xrightarrow{\beta} p'$. We denote the set of conformance failures of p by $\mathrm{F}_{cf}(p)$.*

This definition in fact generalizes secure traces:

Lemma 4. *$\alpha \in \mathrm{Tr}_{cf}(p)$ if and only if for all prefix $\beta \leq \alpha$ with $\alpha = \beta \cdot a \cdot \beta'$ we have $\langle \beta, \{a\} \rangle \in \mathrm{F}_c(p)$.*

Proof. We use the fact that $\alpha \in \mathrm{Tr}_{cf}(p)$ if and only if $\beta \in \mathrm{Tr}_{cf}(p)$ for all prefix $\beta \leq \alpha$.

- For $\alpha = \langle\rangle$ we always have that $\langle\rangle \in \mathrm{Tr}_{cf}(p)$ and, obviously, for all $\beta \cdot a \cdot \beta' = \alpha$ we have $\langle \beta, \{a\} \rangle \in \mathrm{F}_{cf}(p)$ since there is no such decomposition.
- For $\alpha = \beta \cdot \langle a \rangle$ we have $\alpha \in \mathrm{Tr}_{cf}(p)$ if and only if $p \xrightarrow{\alpha}$ and $(p \xrightarrow{\beta} p' \Longrightarrow p' \xrightarrow{a})$. We also have $\langle \beta, \{a\} \rangle \in \mathrm{F}_{cf}(p)$ if and only if $(p \xrightarrow{\beta} p' \Longrightarrow p' \xrightarrow{a})$, so the left to right part of the result is obvious, since $\mathrm{Tr}_{cf}(p)$ is closed by prefixes. To conclude the implication from right to left, we only need to derive $p \xrightarrow{\alpha}$ from the hypothesis. This can be proved by checking that we have $p \xrightarrow{\beta}$ for all $\beta \leq \alpha$ by induction on the lenght of β:
  - $\beta = \langle\rangle$ is trivial.
  - If we have $\beta = \beta \cdot \langle a \rangle$, by induction hypothesis we can assume that $p \xrightarrow{\beta'}$ and then, we have $p \xrightarrow{\beta'} p'$ and by applying the hypothesis we also have $p' \xrightarrow{a}$, which gives us $p \xrightarrow{\beta}$. ☐

7 Conclusions

We have presented appropriate versions of the most well-known linear semantics to fit within the two frameworks that we have developed to take into account the (desired) contravariant character of some of the elements that conform the behaviour of systems. In more detail, we have considered the more general covariant-contravariant framework, based on covariant-contravariant simulations, and the conformance framework, based on the more specific conformance simulations.

We have shown that in that second case only the notion of trace needs a revised definition, that of secure traces, which are those whose actions can never be refused along any of their executions. For the remaining classic linear semantics, no substantial changes are needed in order to obtain their conformance versions. Actually, it could be said that they indeed need a radical revision because if we wanted to use set inclusion between the corresponding sets of observations (failures, readiness, ...) then we would need to take the complements (no-failures, no-readiness, ...) of the observations in the plain (covariant) case. However, we could also argue that this means that the essence of the semantics remains the same so that the classic linear semantics based on the observation of the ready sets are also suitable for the conformance framework.

The picture is completely different when the general covariant-contravariant case is considered. We have defined both covariant-contravariant traces and failures in a natural manner but, even so, the proofs of their properties are more complex than we expected. Moreover, the definition of the (reasonable) covariant-contravariant versions of the rest of the classic linear semantics (readiness, failures traces, and ready traces) seems to be complicated. We have shown that the $\exists\forall$-nesting generated by the covariant-contravariant treatment of the different kinds of actions has to be taken into account; now, still calling "linear" to the obtained semantics becomes a bit unnatural since they are based on a game-like tree construction. We envisage a relation with the known-game semantics [1, 2]; this direction deserves further exploration.

Our main conclusion is that both the simulation framework and the coarsest linear semantics at each level of the extended ltbt-spectrum [12] have been proved to be specially robust. The definitions of the covariant-contravariant versions of those semantics (ready simulation, traces, failures) have come out quite naturally. This agrees to some extent to what we have in practice, where most of the developments that have happened are either based on branching-time semantics (simulation, bisimulation, ready simulation) or on those coarsest linear semantics. Once we have established the foundations of these generalized covariant-contravariant semantics, we will look for concrete applications in which to use them to specify properties of those systems in which we find together both covariant and contravariant features.

References

1. S. Abramsky. Algorithmic game semantics: A tutorial introduction. In *Proof and System Reliability, Proceedings of the NATO Advanced Study Institute, Marktoberdorf*, pages 21–47. Kluwer Academic Publishers, 2001.
2. S. Abramsky and G. McCusker. Game semantics. In *Computational Logic: Proceedings of the 1997 Marktoberdorf Summer School*, pages 1–56. Springer-Verlag, 1999.
3. L. Aceto, I. Fábregas, D. de Frutos Escrig, A. Ingólfsdóttir, and M. Palomino. Relating modal refinements, covariant-contravariant simulations and partial bisimulations. In FSEN 2011, LNCS. Springer, 2011.
4. L. Aceto, A. Ingolfsdottir, P. B. Levy, and J. Sack. Characteristic formulae for fixed-point semantics: A general framework. *Mathematical Structures in Computer Science, special issue devoted to selected papers from EXPRESS 2009*, 2011.
5. D. de Frutos Escrig, C. Gregorio-Rodríguez, and M. Palomino. On the unification of semantics for processes: observational semantics. In *SOFSEM 09*, volume 5404 of *LNCS*, pages 279–290. Springer, 2009.
6. I. Fábregas, D. de Frutos-Escrig, and M. Palomino. Non-strongly stable orders also define interesting simulation relations. In *CALCO 09*, volume 5728 of *LNCS*, pages 221–235. Springer, 2009.
7. I. Fábregas, D. de Frutos-Escrig, and M. Palomino. Equational characterization of covariant-contravariant simulation and conformance simulation semantics. In *Proceedings Seventh Workshop on Structural Operational Semantics*, volume 32 of *Electronic Proceedings in Theoretical Computer Science*, pages 1–14, 2010.
8. I. Fábregas, D. de Frutos-Escrig, and M. Palomino. Logics for contravariant simulations. In *FMOoDS-FORTE 2010*, volume 6117 of *LNCS*, pages 224–231. Springer, 2010.
9. K. G. Larsen, U. Nyman, and A. Wasowski. Interface input/output automata. In *FM*, volume 4085 of *LNCS*, pages 82–97. Springer, 2006.
10. G. Leduc. A framework based on implementation relations for implementing LOTOS specifications. *Computer Networks and ISDN Systems*, 25(1):23–41, 1992.

11. N. Lynch. I/o automata: A model for discrete event systems. *In 22nd Annual Conference on Information Sciences and Systems*, pages 29–38, 1988.
12. R. J. van Glabbeek. The linear time-branching time spectrum I: The semantics of concrete, sequential processes. In *Handbook of process algebra*, pages 3–99. North-Holland, 2001.