# A logical characterisation for input output conformance simulation ioco$\underline{s}$*


Luca Aceto, Ignacio Fábregas, Carlos Gregorio-Rodríguez, and Anna Ingólsfdóttir

Departamento Sistemas Informáticos y Computación
Universidad Complutense de Madrid, Spain

ICE-TCS, School of Computer Science
Reykjavik University, Iceland.


## 1 Introduction

Over the last couple of years we have been studying the input-output conformance simulation relation (ioco$\underline{s}$) [5, 8, 6, 7] that refines the classic input-output conformance testing (ioco) theory due to Tretman. The work by Tretmans [10] has provided a widely used and theoretically well-founded framework for the Model Based Testing (MBT) community: it offers both offline and online testing algorithms [3], and there are several model-based test generation tools that implement the ioco-testing theory.

From a theoretical point of view, some interesting features of the ioco-framework are as follows: behaviours are modelled as labelled transition systems (LTS); quiescent states (see [9]) are considered; implementations should be input enabled; and the ioco relation is a trace-based semantics, and thus a linear semantics [11].

The ioco$\underline{s}$ approach also considers LTS as models, quiescence, and shares much of the ioco philosophy, but it considers a wider domain of behaviours, not imposing, but allowing, implementations to be input enabled. The substantial difference between the two approaches is that the conformance relation underlying ioco$\underline{s}$ is an input-output simulation (a branching-time semantics [11]) with greater discriminatory power than ioco (see [6, Theorem 1]).

Simulation is an important notion pervading many fields in computer science (model checking, concurrency theory, formal verification...), with a plethora of theoretical and practical applications. For example, results presented in [6], indicate that ioco$\underline{s}$ may be used to minimise LTSs in model checking as a technique to alleviate the state explosion problem.

In more detail, ioco$\underline{s}$ is a simulation-based semantics over LTSs developed under the assumption that systems have two kinds of transitions: input actions, those that the systems are willing to admit or respond to, and output actions, those produced by the system and that can be seen as responses or results. We call $I$ the alphabet for input actions and write $a?, b?, c?\ldots$ for typical members of $I$. We denote with $O$ the alphabet for output actions and use $x!, y!, \delta!\ldots$ ro range over $O$.

A state with no output actions cannot autonomously proceed; such a state is called *quiescent*. For the sake of simplicity and without loss of generality (see for instance [10, 9]), we directly introduce the event of quiescence as a special action denoted by $\delta! \in O$ into the definition of our models.

The formal definition of ioco$\underline{s}$ considers the following functions on states of labelled transition systems:


*Research partially supported by the Spanish MEC projects TIN2012-36812-C02-01 and TIN2012-39391-C04-04, the project 001-ABEL-CM-2013 within the NILS Science and Sustainability Programme and the project *Nominal SOS* (project nr. 141558-051) of the Icelandic Research Fund

$$\mathrm{outs}(p) = \{o! \mid o! \in O,\ p \xrightarrow{o!}\}, \text{ the set of initial outputs of a state } p.$$
$$\mathrm{ins}(p) = \{a? \mid a? \in I,\ p \xrightarrow{a?}\}, \text{ the set of initial inputs of a state } p.$$

**Definition 1.** *We say that a binary relation $R$ of states in a labelled transition system is a* $\mathrm{ioco}\underline{s}$*-relation if and only if for any $(p, q) \in R$ the following conditions hold:*

1. $\mathrm{ins}(q) \subseteq \mathrm{ins}(p)$
2. $\forall a? \in \mathrm{ins}(q)$ *if* $p \xrightarrow{a?} p'$ *then* $\exists q'$ *such that* $q \xrightarrow{a?} q' \wedge (p', q') \in R$.
3. $\forall o! \in \mathrm{outs}(p)$ *if* $p \xrightarrow{o!} p'$ *then* $\exists q'$ *such that* $q \xrightarrow{o!} q' \wedge (p', q') \in R$.

*We define the* input-output conformance simulation *(*$\mathrm{ioco}\underline{s}$*) as the union of all* $\mathrm{ioco}\underline{s}$*-relations (the biggest* $\mathrm{ioco}\underline{s}$*-relation). We will denote by* $\mathrm{ioco}\underline{s}_{\equiv}$ *the kernel of the* $\mathrm{ioco}\underline{s}$ *preorder.*

## 2 Contribution: Logic for $\mathrm{ioco}\underline{s}$

We present for the first time a logical characterization of the $\mathrm{ioco}\underline{s}$ relations, both the preorder and equivalence. This logic is a non-standard subset of Hennessy-Milner Logic and is rather *minimal* although convenient to characterize clearly the discriminating power of the $\mathrm{ioco}\underline{s}$ relation.

**Definition 2.** *The syntax of the logic for* $\mathrm{ioco}\underline{s}$*, denoted by* $\mathcal{L}_{\mathrm{ioco}\underline{s}}$*, is defined by the following grammar.*

$$\phi ::= \mathbf{tt} \mid \mathbf{ff} \mid \phi \wedge \phi \mid \phi \vee \phi \mid \langle\!| a? |\!\rangle \phi \mid \langle x! \rangle \phi,$$

*where $a? \in I$ and $x! \in O$. The semantics of the atomic propositions* $\mathbf{tt}$ *and* $\mathbf{ff}$ *and of the Boolean connectives $\wedge$ and $\vee$ is defined as usual. The modalities $\langle\!| a? |\!\rangle$ and $\langle x! \rangle$, are defined as follows:*

- $p \models \langle x! \rangle \phi$ *iff* $p' \models \phi$ *for some* $p \xrightarrow{x!} p'$.
- $p \models \langle\!| a? |\!\rangle \phi$ *iff* $p \not\xrightarrow{a?}$ *or* $p' \models \phi$ *for some* $p \xrightarrow{a?} p'$.

It is well-known that every logic naturally induces a preorder on a given set of processes given by: $p \leq_{\mathcal{L}} q$ iff $\forall \phi \in \mathcal{L}\ \ p \models \phi$ then $q \models \phi$. Hence, the logic $\mathcal{L}_{\mathrm{ioco}\underline{s}}$ induces the preorder $\leq_{\mathcal{L}_{\mathrm{ioco}\underline{s}}}$. The main contribution is that this logical preorder coincides with the $\mathrm{ioco}\underline{s}$ relation. That is, we have:

**Theorem 2.1** (Logical characterization for $\mathrm{ioco}\underline{s}$)**.** *$p\ \mathrm{ioco}\underline{s}\ q$ iff $\forall \phi \in \mathcal{L}_{\mathrm{ioco}\underline{s}}\quad p \models \phi$ then $q \models \phi$.*

**Corollary 2.2.** *$p\ \mathrm{ioco}\underline{s}_{\equiv}\ q$ iff ($\forall \phi \in \mathcal{L}_{\mathrm{ioco}\underline{s}}\quad p \models \phi$ iff $q \models \phi$).*

**Corollary 2.3.** *For all $\phi$ in $\mathcal{L}_{\mathrm{ioco}\underline{s}}$ if we want to check $p \models \phi$, it is equivalent to minimise $p$ to $q$ (using the generalized coarsest partitioning algorithm from [12, 6] and decide whether $q \models \phi$.*

Finally, applying the results in [1] we cen define the characteristic formula for each process in a finite LTS modulo the $\mathrm{ioco}\underline{s}$ preorder.

## 3 Future work

It seems natural to compare the previous logics for $\mathrm{ioco}\underline{s}$ with similar logics that are already in the literature. In particular, we find it interesting to explore its relation with the logics for ready simulation [11, 2], covariant-contravariant simulation and conformance simulation [4]. Since the study of property preservation for expressive logics is of great interest for the model checking community, we also plan to study how the properties preserved by $\mathrm{ioco}\underline{s}$ are related with those expressible in fragments of Action Based CTL and of the $\mu$-calculus.

## References


[1] Luca Aceto, Anna Ingólfsdóttir, and Joshua Sack. Characteristic formulae for fixed-point semantics: A general framework. In Sibylle B. Fröschle and Daniele Gorla, editors, *Proceedings 16th International Workshop on Expressiveness in Concurrency, EXPRESS 2009, Bologna, Italy, 5th September 2009.*, volume 8 of *EPTCS*, pages 1–15, 2009.

[2] David de Frutos-Escrig, Carlos Gregorio-Rodríguez, Miguel Palomino, and David Romero-Hernández. Unifying the linear time-branching time spectrum of process semantics. *Logical Methods in Computer Science*, 9(2:11):1–74, 2013.

[3] René G. de Vries and Jan Tretmans. On-the-fly conformance testing using spin. *STTT*, 2(4):382–393, 2000.

[4] Ignacio Fábregas, David de Frutos-Escrig, and Miguel Palomino. Logics for contravariant simulations. In *FMOODS-FORTE 2010*, volume 6117 of *Lecture Notes in Computer Science*, pages 224–231. Springer, 2010.

[5] Carlos Gregorio-Rodríguez, Luis Llana, and Rafael Martínez-Torres. Input-output conformance simulation (iocos) for model based testing. In Dirk Beyer and Michele Boreale, editors, *FMOODS/FORTE*, volume 7892 of *Lecture Notes in Computer Science*, pages 114–129. Springer, 2013.

[6] Carlos Gregorio-Rodríguez, Luis Llana, and Rafael Martínez-Torres. Effectiveness for input output conformance simulation iocos. In Erika Ábrahám and Catuscia Palamidessi, editors, *FORTE*, volume 8461 of *Lecture Notes in Computer Science*, pages 100–116. Springer, 2014.

[7] Carlos Gregorio-Rodríguez, Luis Llana, and Rafael Martínez-Torres. Extending mcrl2 with ready simulation and ioco$\underline{s}$ input-output conformance simulation. In *SAC-SVT to appear*, 2015.

[8] Luis Llana and Rafael Martínez-Torres. IOCO as a simulation. In Steve Counsell and Manuel Núñez, editors, *Software Engineering and Formal Methods - SEFM 2013 Collocated Workshops: BEAT2, WS-FMDS, FM-RAIL-Bok, MoKMaSD, and OpenCert, Madrid, Spain, September 23-24, 2013, Revised Selected Papers*, volume 8368 of *Lecture Notes in Computer Science*, pages 125–134. Springer, 2013.

[9] Gerjan Stokkink, Mark Timmer, and Mariëlle Stoelinga. Talking quiescence: a rigorous theory that supports parallel composition, action hiding and determinisation. In Alexander K. Petrenko and Holger Schlingloff, editors, *MBT*, volume 80 of *EPTCS*, pages 73–87, 2012.

[10] Jan Tretmans. Model based testing with labelled transition systems. In Robert M. Hierons, Jonathan P. Bowen, and Mark Harman, editors, *Formal Methods and Testing*, volume 4949 of *Lecture Notes in Computer Science*, pages 1–38. Springer, 2008.

[11] Rob J. van Glabbeek. *Handbook of Process Algebra*, chapter The Linear Time – Branching Time Spectrum I: The Semantics of Concrete, Sequential Processes, pages 3–99. Elsevier, 2001.

[12] Rob J. van Glabbeek and Bas Ploeger. Correcting a space-efficient simulation algorithm. In Aarti Gupta and Sharad Malik, editors, *CAV*, volume 5123 of *Lecture Notes in Computer Science*, pages 517–529. Springer, 2008.