A General Theory of Barbs, Contexts and Labels

FILIPPO BONCHI, ENS Lyon, Université de Lyon, LIP (UMR 5668 CNRS ENS Lyon UCBL INRIA)
FABIO GADDECCI, Dipartimento di Informatica, Università di Pisa
GIACOMA VALENTINA MONREALE, Dipartimento di Informatica, Università di Pisa

Barbed bisimilarity is a widely-used behavioural equivalence for interactive systems: given a set of predicates (denoted “barbs” and representing basic observations on states) and a set of contexts (representing the possible execution environments), two systems are deemed to be equivalent if they verify the same barbs whenever inserted inside any of the chosen contexts. Despite its flexibility and expressiveness, this definition of equivalence is unsatisfactory, since often the quantification is over an infinite set of contexts, thus making barbed bisimilarity very hard to be verified.

Should a labelled operational semantics be available, more efficient observational equivalences might be adopted. To this end, a series of techniques have been proposed to derive labelled transition systems (LTSs) from unlabeled ones, the main example being Leifer and Milner’s theory of reactive systems. The underlying intuition is that labels should be the “minimal” contexts that allow for a reduction step to be performed.

However, minimality is difficult to assess, while the set of “intuitively” correct labels is often easily devised by the ingenuity of the researcher. This paper introduces a framework that characterises (weak) barbed bisimilarity via LTSs whose labels are (not necessarily minimal) contexts. Differently from previous proposals, our theory is not dependent on the way the labelled transitions are built, and it relies on a simple set-theoretical presentation for identifying those properties such an LTS should verify in order to (1) capture the barbed bisimilarities of the underlying system and (2) ensure that such bisimilarities are congruences.

Furthermore, we adopt suitable proof techniques in order to make feasible the verification of such properties. To provide a test-bed for our formalism, we instantiate it by addressing the semantics of the Mobile Ambients calculus, recasting its barbed bisimilarities via label-based behavioural equivalences.

Categories and Subject Descriptors: C.2.2 [Computation by Abstract Devices]: Modes of Computation
General Terms: Theory
Additional Key Words and Phrases: Barbed bisimilarity, contexts-as-labels paradigm, interactive systems.

1. INTRODUCTION

Nowadays, the dynamics of a computational system is specified operationally by means of a reduction system: a set, often the carrier of an algebra, representing the possible states of the device; and a relation among these states, usually inductively defined with respect to the algebraic structure of states, representing the possible evolutions of the device. Despite the advantage of conveying the operational semantics with relatively few compact rules, freely instantiated, the main drawback of reduction-based solutions is poor compositionality, since the dynamic behaviour of standalone terms can be interpreted only by inserting them in appropriate contexts, where a reduction may take place. Indeed, this is the core notion underlying barbed bisimilarity, possibly the most used behavioural equivalence for interactive systems: given a set of boolean predicates (denoted “barbs” and representing basic observations on states) and a set of contexts
(representing the admissible execution environments), two systems are deemed to be equivalent if they verify the same barbs whenever inserted inside any of the chosen contexts. Despite its flexibility and expressiveness, this definition of equivalence is often unsatisfactory, since the quantification may be over an infinite set of contexts, thus making barbed bisimilarity very hard to be verified.

Should a labelled operational semantics be available, more efficient observational equivalences could be adopted. To this end, a series of techniques have been proposed to derive labelled transition systems (LTSs) from unlabeled ones. Possibly, the most successful solution proposed for the conundrum is the context-as-label paradigm, as pioneered for process calculi by Sewell [Sewell 2002]. The idea is simple: whenever a system specified by a term \( C[P] \), i.e., by a sub-term \( P \) inserted into a (unary) context \( C[\cdot] \), may evolve to a term \( Q \), the associated LTS has a transition from \( P \) to \( Q \) labelled by \( C[\cdot] \), i.e., the state \( P \) evolves into the state \( Q \) with an observation \( C[\cdot] \). Now, two systems are deemed equivalent if they are bisimilar in the derived LTS.

Of course, such a technique is meaningful as long as the set of labels is suitably chosen. At least, the set should be finite and somehow irredundant, in order to perform as little checks as possible. Also, the associated bisimilarity should be a congruence. Indeed, these two notions are at the core of the reactive system approach proposed by Leifer and Milner [Leifer and Milner 2000]. The key notion is to have a (suitably defined) notion of minimal context and to adopt only those as labels, and the main result is that bisimilarity for the derived LTS is indeed a congruence. The work has been further generalized, starting e.g. from conditional reactive systems [Hu"ulbsch and K"onig 2012] or considering weak bisimilarities [Bruni et al. 2005]. Most important was the notion of borrowed context by Ehrig and K"onig [Ehrig and K"onig 2006], instantiating to graph transformation the notion of minimality: it represents one of the few algorithmic way of distilling minimal labels, and indeed, it proved fruitful in tackling successful case studies (see e.g. [Bonchi et al. 2009a] and the references therein).

The authors of the present paper also worked on extending the reactive systems formalisms by tackling barbed bisimilarities, and considering when such similarities are captured by the LTS whose labels are minimal contexts according to Leifer and Milner’s theory. The key feature of minimality is also a burning issue, given the inherent difficulty of assessing it. Technically, this has to be addressed by resorting to rather sophisticated categorical structures such as quasi- and bi-categories [Milner 2006; Sassone and Sobociński 2003]. This difficulty represents a main problem, more so given the often intuitive nature of those labels: instead of identifying them, the whole issue is to actually prove the minimality of the intuitively correct contexts, and the burden is exacerbated by the need to resort to a rather complex categorical machinery.

This paper proceeds in an orthogonal way. Instead of a descriptive approach, stating how the labels should be shaped, we take a prescriptive one, and we look for properties of an LTS whose labels are not necessarily minimal contexts such that (1) the LTS captures the barbed bisimilarity of the underlying RS, and (2) its associated bisimilarity is a congruence. Furthermore, we develop some proof techniques in order to ease the burden of proving these properties. The result is a streamlined framework into which the key properties (1) and (2) can be easily checked. Indeed, the minimal context LTS is just one of those verifying the properties above, and most often, the technical burden of proving minimality is lifted by the choice of the “intuitively” correct labels. Indeed, the recent results on bisimilarities for concurrent constraint programming [Aristizabal et al. 2011] suggest that many examples are just waiting to be further explored. To a certain extent, we rely on the ingenuity of the researchers, as suggested also in the original papers by Sewell, starting the whole contexts-as-labels issue. We just offer a framework where the key issues of such an intuition may be safely and easily verified.
The paper has the following structure. In Section 2 we introduce our variant of reactive systems, followed by three sub-sections on the running examples (both synchronous and asynchronous CCS [Milner 1989]), on the definition of barbed saturated semantics (by means of barbs, contexts and unlabeled reductions), and on some pruning techniques for such bisimilarities. In Section 3 we introduce context LTSs, i.e., LTSs having contexts as labels, identifying the key notions of soundness and active completeness. Two proof techniques exploiting those sound and actively complete LTSs to develop efficient techniques for proving barbed saturated semantics are then introduced: they are semi-saturated and \( L \)-bisimilarities, presented in Sections 4 and 5, respectively. Finally, Section 6 illustrates how to apply our techniques to the semantics of the calculus of Mobile Ambients [Cardelli and Gordon 2000].

Sources The present work comes as the concluding chapter of a series of papers devoted to barbed semantics for reactive systems. As such, it relies and largely extends previous results, in particular those appearing on [Bonchi et al. 2009b] for barbed semantics of reactive systems, on [Bonchi et al. 2009d] for the introduction of \( L \)-bisimilarity, and on [Bonchi et al. 2011] for the general framework characterising barbed bisimilarity via transition systems whose labels are not necessarily minimal contexts. More precisely, in this paper we enrich the theory in [Bonchi et al. 2011] in several ways. First of all, our framework now encompasses a much larger class of LTSs, the sound and actively complete ones. Second, we introduce a novel pruning technique for the barbed saturated semantics, in order to increase its degree of feasibility. Finally, we recast the results in [Bonchi et al. 2009b] and [Bonchi et al. 2009d] to this general framework. We also include detailed proofs of most of the results. Those missing are either considered immediate or analogous to other proofs in the paper.

2. REACTIVE SYSTEMS

This section introduces a framework that aims to be a general theory for modeling the strong and weak (barbed) semantics of interactive systems. As running examples and main case study, we will show how it encompasses a few process calculi notation that is adopted in later sections.

Definition 2.1 A monoid is a triple \( \mathbb{M} = (M, \otimes, 1) \) such that \( M \) is a set, \( \otimes : M \times M \to M \) is an associative binary operator, and \( 1 \in M \) is the identity element.

Given a set \( X \), a monoid action of \( \mathbb{M} \) on \( X \) is an operation \( \cdot : M \times X \to X \) compatible with the monoid operation, i.e., such that for each \( m_1, m_2 \in M \) and \( x \in X \), \( m_1 \cdot (m_2 \cdot x) = (m_1 \otimes m_2) \cdot x \) and for each \( x \in X \), \( 1 \cdot x = x \).

Definition 2.2 (System Theory). A system theory is a triple \( S = (P, C, \cdot) \) such that \( P \) is a set of processes, ranged over by \( P, Q, R, \ldots \), \( C = (C, \cdot, \cdot) \) a monoid of contexts, ranged over by \( C[-], D[-], \ldots \), and \( \cdot : C \times P \to P \) a monoid action.

We usually denote context composition \( C_1[-] \circ C_2[-] \) as \( C_2[C_1[-]] \) and the action \( C[-] \cdot P \) as \( C[P] \). The chosen notation supports the intuition that the monoid operation represents the functional composition of unary contexts, while the action is just the insertion of a process into a context. It allows for an easier comparison with the process calculi notation that is adopted in later sections.

Definition 2.3 (Reactive System). A reactive system is a triple \( R = (S, \leadsto, O) \) such that \( S \) is a system theory, \( \leadsto \subseteq P \times P \) a relation and \( O \) a set of predicates on \( P \).

We write \( P \leadsto Q \) to mean \( (P, Q) \in \leadsto \), and we denote by \( \leadsto^* \) the reflexive and transitive closure of the reduction relation \( \leadsto \). The predicates in \( O \) are called barbs and they represent basic observations on the states of a system. We write \( P \Downarrow_o \) if \( P \) satisfies
### 2.1. Running example

In order to illustrate our approach, we introduce both the synchronous and asynchronous variants of CCS [Milner 1989]. More precisely, we take the finite, restriction-free fragment of CCS with the reduction semantics shown in [Milner 1999] and its asynchronous counterpart. However, our considerations extend to the full calculus.

The syntax of CCS is defined by the grammar on the left of Fig. 1, the one for asynchronous CCS (ACCS) by the grammar on the right. In both cases, we assume a set $N$ of names, ranged over by $a, b, c, \ldots$, with $\tau \notin N$. We let $P, Q, R, \ldots$ range over the set of processes and $M, N, O, \ldots$ over the set of summations. We define $P_s$ and $P_A$ as the set of CCS and ACCS processes taken up-to structural congruence $\equiv$. This e.g. means that we usually omit the brackets for associativity of the parallel and non-deterministic operators. Note that also the different operational semantics we are going to present for those calculi are also defined up-to structural congruence: this will allow for simpler presentations, and it is actually a necessary requirement for reduction semantics.

The transition relation $\rightarrow_S$ for CCS is defined by rules SYN, TAU, and PAR. In particular, rule SYN allows a process $a.P + M$ that is ready to receive an input on $a$ to synchronise with a process $\bar{a}.Q + N$ that is ready to send an output on the same channel. For the asynchronous calculus, the reduction relation $\rightarrow_A$ is obtained by replacing the rule SYN by ASYN: the occurrence of an unguarded $\bar{a}$ indicates a message that is available on some communication media named $a$. The message disappears after it is received. Output prefixes $\bar{a}.P$ are absent in ACCS, the intuition being that message sending is non-blocking and thus the reception of a message cannot enable a continuation.

The notion of barb for CCS has been introduced in [Milner and Sangiorgi 1992]. In the synchronous calculus, a process has an input (output) barb on $a$ if it is ready to perform an input (output) on $a$. Formally, if $\alpha \in \{a, \bar{a}\}$, then $P \downarrow_{\alpha}$ when $P \equiv \alpha.P_1 + M|P_2$ for some processes $P_1, P_2$ and summation $M$.

In the asynchronous case instead, only output barbs are considered [Amadio et al. 1998], defined by $P \downarrow_{\bar{a}}$ when $P \equiv \bar{a}P_1$ for some process $P_1$. The idea is that, since message sending is non-blocking, an external observer can just send messages without knowing if they will be received or not. Hence, inputs are deemed unobservable. Hereafter, we will use $O_s$ and $O_A$ to denote the set of barbs for the synchronous and asynchronous variant, respectively.

![Fig. 1. The syntax and the reduction semantics of CCS and ACCS.](image-url)

### 2.2. Example

In [Milner and Sangiorgi 1992], the authors consider the system theory $S_S = \langle P_s, C_S, \cdot \rangle$, where $C_S$ is the monoid $\langle C_s, \circ, \_ \rangle$, with $C_s$

---

---
the set of unary CCS contexts (up to structural congruence) generated by the grammar
\[ C[-] := -, C[-]R, \alpha.C[-] + M \]
for \( \alpha \in \{a, \bar{a}, \tau\} \) and \( R, M \) ranging over synchronous processes and summations, respectively. The system theory \( S_A \) together with the reduction relation \( \sim \) and the set of barbs \( O_S \) described above forms the reactive system \( R_A = (S_A, \sim, O_A) \) modeling CCS.

Analogous considerations hold for \( \text{ACCS} \). In this case, the system theory is \( S_A = (P_A, C_A, \cdot) \), where \( C_A \) is the monoid \( (C_A, \circ, -) \), with \( C_A \) the set of unary \( \text{ACCS} \) contexts generated by the grammar \( \dagger \) where \( \alpha \in \{a, \tau\} \) and \( R \) and \( M \) range over asynchronous processes and summations. The RS modeling \( \text{ACCS} \) is therefore \( R_A = (S_A, \sim_A, O_A) \).

2.2. Barbed Saturated Bisimulation

With the ingredients offered by our theory, we can immediately define a behavioural equivalence, equating two processes if these cannot be distinguished by an observer that, in any moment of their execution, can insert them into any context and observe the barbs and the possible reductions.

**Assumption:** In the following, we fix a monoid of contexts \( C = (C, \circ, -) \), a system theory \( S = (P, C, \cdot) \), and a reactive system \( R = (S, \sim, O) \).

**Definition 2.4 (Barbed Saturated Bisimilarity).** A symmetric relation \( B \subseteq P \times P \) is a barbed saturated bisimulation if whenever \( P \sim Q \) then \( \forall C[-] \in C \)

1. if \( C[P] \Downarrow_o \) then \( C[Q] \Downarrow_o \);
2. if \( C[P] \leadsto P' \) then \( C[Q] \leadsto Q' \) and \( P'BQ' \).

The largest barbed saturated bisimulation is called barbed saturated bisimilarity and denoted by \( \sim_{BS} \). Weak barbed saturated bisimilarity and weak barbed saturated bisimilarity \( (\equiv_{BS}) \) are defined as above by replacing \( \sim \) and \( \Downarrow_o \) with \( \sim^* \) and \( \Downarrow_o \).

Both \( \sim_{BS} \) and \( \equiv_{BS} \) are clearly congruences. Moreover, barbed saturated bisimilarity encompasses the standard behavioural equivalences of many calculi. As far as our running example is concerned, asynchronous bisimilarity coincides with barbed saturated bisimilarity for \( \text{ACCS} \) [Amadio et al. 1998] and, analogously, the standard bisimilarity for CCS coincides with the barbed saturated one: we tackle the correspondence issue in Sections 5.1.2 and 5.2.2. We refer instead to Appendix A for a comparison with the original definition of barbed congruence [Milner and Sangiorgi 1992].

The main drawback of this kind of definition is the quantification over all contexts that makes hard the proofs of equivalence. Indeed, this work aims at providing general proof techniques for \( \sim_{BS} \) and \( \equiv_{BS} \) that avoid such quantifications.

We conclude this section presenting a few processes that are (weakly) saturated bisimilar in \( \text{ACCS} \), while they are not so in CCS.

**Example 2.5.** In \( \text{ACCS} \) we have that \( a.a \equiv_{BS} 0 \). In the later sections, we will prove this formally by exploiting our proof techniques, but, for the time being, we give some intuition. The idea is that for all contexts \( C[-] \), if \( C[a.a] \leadsto P \) then the reaction is either in the context \( C[-] \) or it is the result of an interaction between \( C[-] \) and \( a.a \). In the former case, \( P \equiv C[a.a] \) (for some context \( C'[-] \)) and the same reaction can happen for \( 0 \), i.e., \( C[0] \leadsto C'[0] \). In the second case, \( a.a \) can interact only if \( C[-] \) contains an output message \( a \), that is \( C[-] \) is of the shape \( a - |R \) for some process \( R \). Observe that this interaction consumes \( a \) and immediately produces a new message \( a \), so that \( P \equiv a.R \). Therefore \( C[0] \Downarrow_o P \) and thus \( C[0] \sim_{BS} P \).

Instead, in CCS we have that \( a.a \not\equiv_{BS} 0 \); indeed \( a.a \Downarrow_o \), but \( 0 \Downarrow_o \). This argument does not work for the asynchronous case since there are no input barbs in \( O_A \). For similar reasons, \( a.a + \tau.0 \sim_{BS} \tau.0 \) in \( \text{ACCS} \) while \( a.a + \tau.0 \not\sim_{BS} \tau.0 \) in CCS.
The previous example shows that $\sim_{BS}$ and $\approx_{BS}$ strongly depend from all the ingredients of our theory (that is, bars, contexts and reductions) and, for this reason, we will write $\sim_{R}^{BS}$ and $\approx_{R}^{BS}$ if we want to make explicit the reactive system $\mathbb{R}$.

2.3. Two pruning techniques for Barbed Saturated Semantics

We now introduce two techniques allowing for pruning some reductions (Section 2.3.1) and some contexts (Section 2.3.2), without changing the (weak) barbed semantics.

2.3.1. Pruning Inactive Reductions. The first technique consists in identifying those reductions whose presence does not really influence $\sim_{BS}$ and $\approx_{BS}$. Given a transition $C[P] \rightsquigarrow P'$, we observe that $P$ can play two different roles: $P$ is inactive, when the reduction takes place in the context $C[\cdot]$ independently from $P$; while $P$ is active whenever the reduction depends on it.

**Definition 2.6 (Types of Reductions).** Let $C[P] \rightsquigarrow P'$ be a reduction. We say that $P$ is inactive in it if $P' = D[P]$ and $\forall Q \in \mathcal{P}, C[Q] \rightsquigarrow D[Q]$; $P$ is active, otherwise.

For instance, $Q$ is not active in $a.0(a.0)Q \rightsquigarrow 0|0|Q$, while instead $a.0|Q$ is.

With this definition, we can give an alternative characterisation of (weak) barbed saturated bisimilarity, which only takes into account active reductions.

**Definition 2.7 (Active Barbed Saturated Bisimilarity).** A symmetric relation $\mathcal{B} \subseteq \mathcal{P} \times \mathcal{P}$ is an active barbed saturated bisimulation if whenever $P \mathcal{B} Q$ then $\forall C[\cdot] \in \mathcal{C}$

1. if $C[P] \downarrow_o$ then $C[Q] \downarrow_o$;
2. if $P$ is active in $C[P] \rightsquigarrow P'$ then $C[Q] \rightsquigarrow Q'$ and $P' \mathcal{B} Q'$.

The largest active barbed saturated bisimulation is called active barbed saturated bisimilarity and denoted by $\sim_{BS}^A$. Active weak barbed saturated bisimulation and active weak barbed saturated bisimilarity ($\approx_{BS}^A$) are defined as above by replacing $\rightsquigarrow$ and $\downarrow_o$ with $\approx$ and $\approx_o$, respectively.

**Proposition 2.8.** $\sim_{BS}^A \approx_{BS}^A$ and $\approx_{BS}^A \approx_{BS}^A$

**Proof.** First observe that $\sim_{BS}^A \subseteq \sim_{BS}^A$, since all the reductions considered in $\sim_{BS}^A$ are taken into account also in $\sim_{BS}^A$. In order to prove that $\sim_{BS}^A \subseteq \sim_{BS}$, we show that the following relation

$$\mathcal{B} = \{(C[P], C[Q]) \mid P \sim_{A}^{BS} Q, C[\cdot] \in \mathcal{C}\}$$

is a barbed saturated bisimulation. Now, $\forall C'[\cdot] \in \mathcal{C}$

1. if $C'[C[P]] \downarrow_o$ then $C'[C[Q]] \downarrow_o$ since $P \sim_{BS}^A Q$;
2. if $C'[C[P]] \rightsquigarrow P'$ then $P$ is either active or inactive. In the latter case, $P' = D[P]$ and $C'[C[Q]] \rightsquigarrow D[Q]$. Since $P \sim_{A}^{BS} Q$, then $P' = D[P] \mathcal{B} D[Q]$. If $P$ is active, then $C'[C[Q]] \rightsquigarrow Q'$ with $P' \sim_{BS}^A Q'$, because $P \sim_{BS}^A Q$.

The proof for the weak case is analogous, but it uses the characterisation of weak barbed saturated bisimulation that is given by Proposition 2.9 below.

**Proposition 2.9.** A symmetric relation $\mathcal{B} \subseteq \mathcal{P} \times \mathcal{P}$ is a weak barbed saturated bisimulation if and only if whenever $P \mathcal{B} Q$ then $\forall C[\cdot] \in \mathcal{C}$

1. if $C[P] \downarrow_o$ then $C[Q] \downarrow_o$;
2. if $C[P] \rightsquigarrow P'$ then $C[Q] \rightsquigarrow Q'$ and $P' \mathcal{B} Q'$.

**Proof.** It is easy to see that if $\mathcal{B}$ is a weak barbed saturated bisimulation according to Definition 2.4, then $\mathcal{B}$ also satisfies the conditions (1) and (2) above.
For the inverse direction, take $B$ to be a symmetric relation that satisfies conditions (1) and (2) above and suppose that $P B Q$.

(1) Condition (1) of Definition 2.4 is immediate from the following.

(2) If $C[P] \leadsto^* P'$ then there exist $P'_1, \ldots, P'_n$ such that $P'_n = P'$ and $C[P] \leadsto P'_1 \leadsto \ldots \leadsto P'_n$. Since $B$ satisfies condition (2) above, there exist $Q'_1, \ldots, Q'_n$ such that $C[Q] \leadsto^* Q'_1 \leadsto^* \ldots \leadsto^* Q'_n$ and $\forall i \in 1, \ldots, n. P'_i B Q'_i$. Thus, $C[Q] \leadsto^* Q'_n$ and $P' = P'_n B Q'_n$.

$\Box$

2.3.2. Pruning Non-Discriminating Contexts. Besides avoiding to consider some reductions, also the set of contexts to be checked can be restricted without changing the semantics.

**Definition 2.10 (Non-Discriminating Context).** Let $E[-] \in C$ be a context. It is non-discriminating if $\forall P \in T, \forall C[-] \in C$ and $\forall o \in O$

(1) if $C[E[P]] \downarrow_o$ then $\forall Q \in T, C[E[Q]] \downarrow_o$;

(2) if $E[P] \leadsto P'$ then $P$ is inactive in it.

For instance, the context $a.-$ is non-discriminating for both CCS and ACCS, since (1) it hides the strong bars of the process that is inserted inside it, and (2) it inhibits its transitions. More generally, all the contexts of the shape $\alpha.C[-] + M$ are non-discriminating for both calculi.

Now, starting from a reactive system $R$ we can build a new reactive system $R'$ by removing some (possibly all) non-discriminating contexts. For example, we can derive a new reactive systems for CCS by taking $C'_S$ to be the set containing the unary CCS contexts (up to structural congruence) generated by the grammar

$$C[-] ::= -, C[-]R.$$

The new system theory is hence $S'_S = \langle P_S, C'_S, \cdot \rangle$, where $C'_S$ is the monoid of contexts $\langle C_S, \cdot, -, \rangle$, while the new reactive system modelling the calculus is $R'_S = \langle S'_S, \leadsto_s, O_S \rangle$. In the same way, we can define a new reactive system $R'_A$ for ACCS.

The following proposition ensures that (strong and weak) ordered and barbed saturated bisimilarity in the new and in the old system coincide.

**Proposition 2.11.** Let $C'$ be a submonoid of $C$, $S' = \langle P, C', \cdot \rangle$ and $R' = \langle S', \leadsto, O \rangle$. If all contexts in $C$ but not in $C'$ are non-discriminating, then $\approx_{R}^{BS} = \approx_{R'}^{BS}$ and $\approx_{R}^{BS} = \approx_{R'}^{BS}$.

**Proof.** Since all the contexts in $C'$ are also in $C$, then $\approx_{R}^{BS} \subseteq \approx_{R'}^{BS}$ and $\approx_{R}^{BS} \subseteq \approx_{R'}^{BS}$.

In order to prove $\approx_{R}^{BS} \subseteq \approx_{R'}^{BS}$, we show that the relation

$$B = \{(C[P], C[Q]) \mid P \approx_{R}^{BS} Q, C[-] \in C\}$$

is a saturated barbed bisimulation for $R$. Let $P \approx_{R}^{BS} Q$ and $D[-] \in C$. Then there are two cases: either $D[C[-]] \in C'$ or $D[C[-]] \notin C'$.

The former case is straightforward. By hypothesis the context $D[C[-]]$ is non-discriminating. Thus, if $D[C[P]] \downarrow_o$ then also $D[C[Q]] \downarrow_o$ (by property (1) of Definition 2.10). If $D[C[P]] \leadsto P'$ then (by property (2) of Definition 2.10) $P' = E'[P]$ (for some $E'[\cdot] \in C$) and $D[C[Q]] \leadsto E'[Q]$. Since $P \approx_{R}^{BS} Q$, then $E'[P] B E'[Q]$.

For the weak case, we proceed exactly in the same way but we use the alternative characterization of weak barbed saturated bisimulation given by Proposition 2.9. $\Box$

3. CONTEXTS AS LABELS

The main objective of our theory is to find a general way to characterise $\approx_{BS}^{BS}$ and $\approx_{BS}^{BS}$ without the quantification over all contexts by mean of “well-behaved” labeled transition systems (LTSs). The intuition is that in these systems labels are contexts,
(TAU) \[
P \sim Q \\
P \xrightarrow{\tau} Q
\]

(RCV) \[
P \equiv a.Q + M|R \\
P \xrightarrow{[\bar{a}.S+N\tau]} Q|R|S
\]

(SND) \[
P \equiv \bar{a}.Q + M|R \\
P \xrightarrow{[a.S+N\tau]} Q|R
\]

(EXTRED) \[
L \sim R \\
P \xrightarrow{[L\tau]} P|R
\]

Fig. 2. The LTS \(\sigma\), for \(L \sim R\) ranging over the two axioms of \(\sim_{\text{CCS}}\).

(NAU) \[
P \sim Q \\
P \xrightarrow{\alpha} Q
\]

(RCV) \[
P \equiv a.Q + M|R \\
P \xrightarrow{[\bar{a}.\alpha]} Q|R
\]

(SND) \[
P \equiv \bar{a}|Q \\
P \xrightarrow{[a.R+M\tau]} Q|R
\]

(EXTRED) \[
L \sim R \\
P \xrightarrow{[L\alpha]} P|R
\]

Fig. 3. The LTS \(\alpha\), for \(L \sim R\) ranging over the two axioms of \(\sim_{\text{ACCS}}\).

more precisely they are the “minimal” contexts that allow a system to react. In this section we formally identify those LTSs that are well-behaved, i.e., that allow us to characterize barbed saturated semantics. First, we introduce some basic notions.

Definition 3.1 (Context LTS). A context LTS (over \(S\)) is a relation \(\delta \subseteq P \times C \times P\).

A transition \(\langle P, C[-], Q \rangle \in \delta\) is usually denoted by \(P \xrightarrow{C[-]} \delta Q\). We write \(P \xrightarrow{C[-]} \delta Q\) whenever there exist \(P'\) and \(Q'\) such that \(P \xrightarrow{\sim^*} P' \xrightarrow{C[-]} \delta Q' \xrightarrow{\sim^*} Q\). When \(\delta\) is either clear from the context or not relevant, we write \(\rightarrow\) and \(\Rightarrow\) in place of \(\rightarrow_\delta\) and \(\Rightarrow_\delta\).

Example 3.2. Let us consider the LTSs \(\sigma\) for CCS and \(\alpha\) for ACCS, shown in Figs. 2 and 3, proposed in [Bonchi et al. 2009a] and [Bonchi et al. 2009d; 2009b], respectively.

The meaning of the rules generating the two LTSs is analogous. The TAU rule models internal computations. The RCV rule models the communication over a channel \(\alpha\), where the output is offered by the context while the input is offered by the process. Vice versa, in the SND rule, the output is provided by the process while the input is provided by the context. Finally, the rule EXTRED represents two rules, one for rule (SYN) and one for (TAU), the axioms of the reduction relation (see Fig. 1): the process \(L\) of the label represents the left hand side of the axiom and the process \(R\) of the target state represents the right hand side. This rule models transitions where the redex is fully offered by the context (intuitively corresponding to reductions where \(P\) is inactive).

Clearly, both CCS and ACCS LTSs are context LTSs.

In order to characterise the class of LTSs on which the labelled semantics for reactive systems is based, we need to introduce the following definitions.

Definition 3.3 (Reactive Context). Let \(C[-] \in C\) be a context. It is reactive if \(\forall P \in \mathcal{P}\) if \(P \sim P'\) then \(C[P] \sim C[P']\).
For instance, the context \( - | P \) is reactive for both CCS and ACCS, while the prefix \( a. - \) is not. For CCS and ACCS, as well as in most process calculi, non-reactive contexts are also non-discriminating (Definition 2.10). An exception is provided by calculi with priorities [Phillips 2008], where the context \( - | P \) is not reactive (it can inhibit reductions) but it is discriminating (it does not satisfy condition (2) of Definition 2.10).

Note that reactive contexts form actually a submonoid: if \( C_1[\cdot] \) and \( C_2[\cdot] \) are reactive, then also \( C_2[C_1[\cdot]] \) is reactive.

**Definition 3.4 (Decomposition Pair).** Let \( P \) be a process, \( C[\cdot] \) a context and \( C[P] \leadsto P' \) a reduction. A pair of contexts \( (C_1[\cdot], C_2[\cdot]) \) with \( C_2[\cdot] \) reactive is a decomposition pair with respect to \( P \), \( C[\cdot] \) and \( C[P] \leadsto P' \) if there exists a process \( P'' \) such that \( P \xrightarrow{C_1[\cdot]} P'' \), \( C_2[C_1[\cdot]] = C[\cdot] \) and \( C_2[P''] = P' \).

In the following, we usually refer to a decomposition pair with respect to a reduction, leaving implicit both process \( P \) and context \( C[\cdot] \).

**Remark 3.5.** As we are going to see, decomposition is a key property for our LTSs. The idea is that each reduction of a reactive system is witnessed by a suitable labelled transition in the associated LTS. Moreover, the label is required to be a reactive context, so that any further reduction occurring in \( P'' \) will also be witnessed.

For instance, consider the CCS process \( P = a.Q + M \) and the context \( C[\cdot] = -[a.R + N.|S] \). A decomposition pair for \( C[P] \leadsto Q|R|S \) is \( (-[a.R + N, -|S]) \). Indeed, it is easy to check that \( P \xrightarrow{-[a.R + N]} Q|R; \) moreover, if we compose \( -[a.R + N] \) with \( -|S \) we exactly obtain \( C[\cdot] \), and finally the composition between \( Q|R \) and \( -|S \) gives \( Q|R|S \).

We can now characterise the class of LTSs we are interested in: they are context LTSs satisfying suitable soundness and completeness properties.

**Definition 3.6 (Sound and (Actively) Complete Context LTS).** A context LTS \( \delta \) is sound if whenever \( P \xrightarrow{[\cdot]} P' \) then \( C[P] \leadsto P' \). It is actively complete if whenever \( C[P] \leadsto P' \) and \( P \) is active in it then the reduction has a decomposition pair. It is complete if it satisfies the completeness property also for possibly inactive \( P \).

In a sound LTS each labelled transition is the witness of a reduction, which in turn by completeness may be witnessed by at least one such labelled transition (according to how many decomposition pairs it has). It is a simple check to verify that both the LTS \( \sigma \) for CCS and the LTS \( \alpha \) for ACCS are sound and complete context LTSs.\(^1\)

Starting from a sound and actively complete LTS, it is possible to derive a smaller one which continues to be actively complete. This can be obtained by removing from the former those labelled transitions, which we will call inactive, intuitively representing the transitions that all processes can perform.

**Definition 3.7 (Types of Transitions).** A transition is said inactive if it has the shape \( P \xrightarrow{[\cdot]} D[P] \) and \( \forall Q \in \mathcal{P} \) we also have \( Q \xrightarrow{[\cdot]} D[Q] \); it is said active otherwise.

**Proposition 3.8.** Let \( \delta \) be a sound and actively complete context LTS. By removing any set of inactive transitions it remains the same (still sound and actively complete).

\(^1\)This is not surprising, since they were distilled in [Bonchi et al. 2009a; Bonchi et al. 2009d] by exploiting Leifer and Milner’s theory and by considering respectively the reactive systems \( \mathbb{R}_a \) and \( \mathbb{R}_a' \) (obtained by removing non-discriminating contexts). Indeed, those LTSs obtained by that theory are both sound (thanks to [Leifer and Milner 2000, Proposition 3]) and complete (thanks to [Leifer and Milner 2000, Proposition 1]).
PROOF. Let δ′ be the derived LTS. Its soundness is obviously guaranteed by the soundness of δ.

We have to show that δ′ is also actively complete, i.e., that for each C[P] ⊢ P′ with P active in it there exists a decomposition pair in δ′.

Since δ is actively complete, by definition there exists a decomposition pair for C[P] ⊢ P′ in δ. This means that there exists ⟨C1[−],C2[−]⟩ such that P  C1[−] ⊢ P′, C2[C1[−]] = C[−] and C2[P′′] = P′. Now, we have to show that this decomposition pair exists also in δ′, that is, P  C1[−] ⊢ P′′ is active and so P  C1[−] ⊢ P′′. We prove this by contradiction. Suppose that P  C1[−] ⊢ P′′ is inactive. This means that P′′ = D[P] and ∀Q ∈ P, Q  C1[−] ⊢ D[Q]. Therefore, thanks to the soundness property, we also have that C1[P] ⊢ P′′ = D[P] and ∀Q ∈ P, C1[Q] ⊢ D[Q]. Moreover, we also know that C[P] ⊢ P′, that is, C2[C1[P]] ⊢ C2[D[P]], and since C2[−] is reactive, ∀Q ∈ P, C2[C1[Q]] ⊢ C2[D[Q]], which by definition means that P is inactive in C[P] ⊢ P′′. □

Let α′ and σ′ be those LTSs obtained respectively from α and σ by removing the transitions generated by the corresponding EXTRED rules. They are no more complete. Consider for example the CCS reduction C[P] ⊢ P′, where P = b.0, C[−] = a.0|a.0|− and P′ = 0|0|P. It is easy to check that there exists no decomposition pair for this reduction. Indeed, the only labelled transition for P is P ⊢ [b,S+N] + N 0|S, but it is obvious that there exists no context C2[−] such that C2[C2[−]] = C[−]. However, thanks to Proposition 3.8, we know that the LTSs α′ and σ′ are actively complete.

In the following, the notion of active completeness will take main stage. The stronger property of completeness is mentioned here for historical reasons, being the one originally introduced in [Bonchi et al. 2011], and for a comparison with Leifer and Milner’s theory, which may be used to generate sound and complete LTSs. We will thus concentrate ourselves in the study of sound and actively complete context LTSs.

Assumption: In the following, we fix a sound and actively complete context LTS δ. Moreover, any context LTS appearing in the rest of the paper will be assumed to have those two properties, without repeating that again and again.

3.1. Decomposable LTSs

The main results of our theory concerns the labelled characterisations of (weak) barbed saturated bisimilarity via semi-saturation (Section 4) and L-bisimilarity (Section 5). These results require that the LTS under consideration is sound and actively complete.

However, it is quite natural to ask under which conditions on a context LTS the standard bisimilarity is a congruence. This is ensured by the notion introduced below.

Definition 3.9 (Universal Decomposition). Let ⟨C1[−],C2[−]⟩ be a decomposition pair for P, C′′[C[−]] and C′′[C[P]] ⊢ P′. It is a universal decomposition pair with respect to P, C[−] and C′′[C[P]] ⊢ P′ if ∀Q, Q  C1[−] Q′′ implies C[Q] C′′[Q] ⊢ C2[Q′′].

As before, the process P and the contexts C[−] and C′[−] are usually denoted implicitly. Consider the CCS process P = a.Q + M and the CCS contexts C[−] = −S and C[−] = −[a.R + N]. The decomposition pair ⟨−[a.R + N], −|S⟩ of C′′[C[P]] ⊢ −σ Q|R|S is universal: for all Q such that Q  −[a.R + N] −|σ Q′′|S it holds Q|S −[a.R + N] −|σ Q′′|S.

Definition 3.10 (Decomposable LTS). A context LTS is decomposable if whenever C[P] C1[−] ⊢ P′ then
(1) if \( P \) is active in \( C'[C[P]] \leadsto P' \) then there exists a universal decomposition pair for \( C'[C[P]] \leadsto P' \).

(2) if \( P \) is inactive in \( C'[C[P]] \leadsto P' \) with \( P' = E[P] \) then \( \forall Q. C[Q] \mathrel{\overset{C[\_]}{\leadsto}} E[Q] \).

Note that condition (1) above implies active completeness. Universality ensures that decomposition pairs (thus the labels on \( \delta \)) are chosen uniformly amongst all processes.

**Proposition 3.11.** Let \( \sim \) be the standard notion of bisimilarity [Milner 1989]. If a context LTS is decomposable then \( \sim \) is a congruence.

The result is an immediate consequence of Proposition 5.3.

### 4. BARBED SEMI-SATURATED BISIMULATIONS

In the previous section we introduced LTSs having contexts as labels. Now we would like to exploit those LTSs to develop efficient techniques for proving barbed saturated semantics. The first attempt would consist in trying with the standard notion of bisimilarity \( \sim \) on these LTSs. However, even if for a decomposable LTS it is a congruence, there is no guarantee that \( \sim \) coincides with \( \sim^{BSS} \). For instance, in \( \text{ACCS} \) we have that

\[
\text{a.} \bar{a} + \tau.0 \sim^{BSS} \tau.0 \quad \text{(Example 2.5), but } \text{a.} \bar{a} + \tau.0 \not\sim \tau.0,
\]

since \( \text{a.} \bar{a} + \tau.0 \) can perform a transition labeled with \( -|\bar{a} \) while \( \tau.0 \) cannot.

This section presents a new proof technique for saturated semantics via a sound and actively complete context LTS: both the strong and the weak cases are discussed. Of course, such a new technique is relevant as much as the saturated semantics are able to capture standard ones for the calculus at hand. As we already remarked, we will deal with this issue for the running examples in Sections 5.1.2 and 5.2.2.

#### 4.1. Strong case

In order to check whether two processes \( P \) and \( Q \) are in a barbed saturated bisimulation (Definition 2.4), one has to verify that they have the same behaviour under all contexts \( C[-] \). This means that (1) \( C[P] \) and \( C[Q] \) expose the same barbs and (2) \( C[P] \) and \( C[Q] \) reduce to processes which are still related. Barbed semi-saturated bisimulation allows to simplify (2), by checking only some of the possible contexts.

**Definition 4.1 (Barbed Semi-Saturated Bisimilarity).** A symmetric relation \( B \subseteq \mathcal{P} \times \mathcal{P} \) is a barbed semi-saturated bisimulation if whenever \( P B Q \) then

1. \( \forall C[-] \in \mathcal{C} \text{ if } C[P] \downarrow_o \text{ then } C[Q] \downarrow_o \);  
2. \( \text{if } P \mathrel{\overset{C[-]}{\downarrow}} P' \text{ then } C[Q] \leadsto Q' \text{ and } P' B Q' \).

The largest barbed semi-saturated bisimulation is called barbed semi-saturated bisimilarity and denoted by \( \sim^{BSS} \).

Reasoning on \( \sim^{BSS} \) is easier than on \( \sim^{BS} \): instead of looking at the reductions under all contexts, we consider only those contexts that actually labels transitions. Still, they capture exactly the same notion of equivalence.

**Proposition 4.2.** \( \sim^{BSS} \subseteq \sim^{BS} \).

**Proof.** We prove that \( \sim^{BSS} \subseteq \sim^{BS} \) by showing that the contextual closure \( S \) of barbed semi-saturated bisimilarity

\[
S = \{ \langle C[P], C[Q] \rangle \mid P \sim^{BSS} Q, \forall C[-] \in \mathcal{C} \}
\]

is a barbed saturated bisimulation. Now, \( \forall C'[\_] \in \mathcal{C} \)

1. If \( C'[C[P]] \downarrow_o \text{ then also } C'[C[Q]] \downarrow_o \), since \( P \sim^{BSS} Q \).
(2) If $C'[C[P]] \sim P'$ then $P$ can be either active or inactive.

In the former case, by active completeness there is a decomposition pair
\( \langle C_1[-], C_2[-] \rangle \) such that $P \sim_{BSS} C_2[C_1[-]] = C'[C[\hat{\alpha}]]$ and $C_2[P] = P'$. Since $P \sim_{BSS} Q$, we have that $C_1[Q] \sim_{BSS} Q''$ with $P'' \sim_{BSS} Q''$. Now, since $C_2[-]$ is reactive, we have that $C_2[C_1[Q]] = C'[C[Q]] \sim_{BSS} C_2[Q']$. Since $P'' \sim_{BSS} Q''$, $P' = C_2[P''] S C_2[Q']$. If $P$ is inactive in $C'[C[P]] \sim P'$ with $P' = D[P]$ then, by definition, $C'[C[Q]] \sim Q'$ with $Q' = D[Q]$. Therefore, since $P \sim_{BSS} Q$, $P' = D[P] S D[Q] = Q'$. 

Now, in order to prove that $\sim_{BS} \subseteq \sim_{BSS}$, it is enough to apply soundness to conclude $C[P] \sim P'$ from $P \sim_{BS} P'$. □

Semi-saturation still requires barbs to be quantified over all contexts. However, for many formalisms (as for CCS and ACCS) it actually suffices to check if $P \perp_o$ implies $Q \perp_o$, since this condition implies that $\forall C[-]$, if $C[P] \perp_o$ then $C[Q] \perp_o$. Barbs satisfying this property are called contextual.

**Definition 4.3 (Contextual Barbs).** A barb $o \in \mathcal{O}$ is contextual if 1) whenever $P \perp_o$ implies $Q \perp_o$ then $\forall C[-] \in \mathcal{C}. C[P] \perp_o$ implies $C[Q] \perp_o$, and 2) whenever $D[P] \perp_o$ then either $P \perp_o$ or $Q. D[Q] \perp_o$.

Via a simple check by induction on the structure of contexts, one can see that the barbs of both CCS and ACCS are contextual.

**Example 4.4.** We use the sound and actively complete LTS $\alpha'$ to formally prove that $a.\bar{a} + \tau.0 \sim_{BS} \tau.0$ in ACCS (as already hinted in Example 2.5). Let $I = P_\Lambda \times P_\Lambda$ be the identity relation and take $\mathcal{B} = \{\langle a.\bar{a} + \tau.0, \tau.0 \rangle, \langle \tau.0, a.\bar{a} + \tau.0 \rangle\} \cup I$. We prove that $\mathcal{B}$ is a semi-saturated bisimulation. We only check the pairs $\langle a.\bar{a} + \tau.0, \tau.0 \rangle$ and $\langle \tau.0, a.\bar{a} + \tau.0 \rangle$, since the other pairs in $I$ are trivial.

(1) Since ACCS barbs are contextual, it is enough to observe that neither $a.\bar{a} + \tau.0$ nor $\tau.0$ satisfies any barb.

(2) The transition $a.\bar{a} + \tau.0 \rightarrow_{\alpha'} 0$ is matched by the reduction $\tau.0 \rightarrow_\Lambda 0$ and clearly $\langle 0, 0 \rangle \in \mathcal{B}$. The transition $a.\bar{a} + \tau.0 \rightarrow_{\lambda} a.\bar{a}$ is matched by the reduction $\tau.0[a] \rightarrow_\Lambda a[\bar{\alpha}]$ and since $0[\bar{\alpha}] = \bar{\alpha}$, then $\langle 0[\bar{\alpha}], \bar{\alpha} \rangle \in I$ (recall that processes in $P_\Lambda$ are taken up to $\equiv$).

In semi-saturated bisimilarity, the transition $P \rightarrow_{\lambda} a.\bar{a}$ should be matched by the reduction $Q[\bar{\alpha}] \rightarrow$. Informally, the latter corresponds to a transition labeled by either $-\bar{a}$ or by the identity context $\cdot$. This is analogous to what happens in [Amadio et al. 1998]: an input transition (intuitively corresponding to a transition with label $-\bar{a}$) can be matched either by an input transition or by a $\tau$ (corresponding to the label $\cdot$). The connection will be further discussed in Section 5.1.2.

### 4.2. Weak case

The results of the previous section extends to the weak semantics with little surprise.

**Definition 4.5 (Weak Barbed Semi-Saturated Bisimilarity).** A symmetric relation $\mathcal{B} \subseteq P \times P$ is a weak barbed semi-saturated bisimulation if whenever $P \mathcal{B} Q$ then

1. $\forall C[-] \in \mathcal{C}$ if $C[P] \perp_o$ then $C[Q] \perp_o$;

2. if $C[P]\perp_o$ then $C[Q] \sim^* Q'$ and $P' \mathcal{B} Q'$.

The largest weak barbed semi-saturated bisimulation is called weak barbed semi-saturated bisimilarity and denoted by $\approx_{BSS}$. 
PROPOSITION 4.6. \( \equiv_{BSS} = \approx_{BS} \).

Proof. Analogous to Proposition 4.2. □

As for the strong case, we can simplify (1) in Definition 4.5 by taking instead

\[(1') \text{ if } P \Downarrow_o \text{ then } Q \Downarrow_o ,\]

whenever the barbs of the system are weak contextual, as defined below.

Definition 4.7 (Weak Contextual Barbs). A barb \( o \in C \) is weak contextual if 1) whenever \( P \Downarrow_o \) implies \( Q \Downarrow_o \), then \( \forall C[-] \in C. C[P] \Downarrow_o \) implies \( C[Q] \Downarrow_o \); and 2) whenever \( D[P] \Downarrow_o \) then either \( P \Downarrow_o \) or \( Q \Downarrow_o \).

Similarly to the strong case, it is easy to prove that the barbs for CCS and ACCS are weak contextual.

Example 4.8. We now prove that \( a.\overline{a} \equiv_{BSS} 0 \) in ACCS, as previously discussed in Example 2.5. Let \( Id = P \times P \) be the identity relation and take \( B = \{ \langle a.\overline{a}, 0 \rangle, \langle 0, a.\overline{a} \rangle \} \cup Id \). We prove that \( B \) is a semi-saturated bisimulation. We only check the pairs \( \langle a.\overline{a}, 0 \rangle \) and \( \langle 0, a.\overline{a} \rangle \), since the other pairs in \( Id \) are trivial.

1) Observe that neither \( a.\overline{a} \) nor \( 0 \) satisfies any weak barb. Moreover, since in ACCS weak barbs are contextual, it holds that \( \forall C[-] \) if \( C[a.\overline{a}] \Downarrow_o \) then \( C[0] \Downarrow_o \), and if \( C[0] \Downarrow_o \) then \( C[a.\overline{a}] \Downarrow_o \).

2) The transition \( a.\overline{a} \xrightarrow{a} a, \overline{a} \) is matched by the reduction \( 0 \xrightarrow{a} 0 \) and, since \( 0 \equiv a, \langle 0, a, \overline{a} \rangle \in Id \).

5. A FAMILY OF EQUIVALENCES: \( L \)-BISIMILARITIES

Despite the fact that, as argued in the previous section, barbed semi-saturated bisimilarity provides an effective proof technique (by avoiding to check reductions of processes inserted into all contexts), it is rather different from standard bisimilarity since (1) a check on barbs is needed and (2) transitions are matched by reductions.

This section introduces \( L \)-bisimilarities, a family of equivalences parametric with respect to a set of contexts \( L \), that is closer to standard bisimilarity \( \sim \) (it avoids (1) and “partially” (2)) but, for some choices of \( L \), characterise barbed saturated \( \sim_{BSS} \). As a corollary, we discover conditions ensuring that \( \sim = \sim_{BSS} \) and, for our running examples, we obtain notions of bisimilarity that are closer to (hence, easier to compare with) those proposed in literature ([Milner 1989] for CCS and [Amadio et al. 1998] for ACCS).

5.1. Strong case

We are now ready to introduce \( L \)-bisimilarity. Intuitively, it can be thought of something in between standard \( \sim \) and semi-saturated \( \sim_{BSS} \) bisimilarity.

Definition 5.1 (\( L \)-Bisimilarity). Let \( L \subseteq C \) be a set of contexts. A symmetric relation \( B \subseteq P \times P \) is an \( L \)-bisimulation if whenever \( P \# B \# Q \) then

\[
\begin{cases}
\text{if } P \xrightarrow{C[-]} P' \text{ then } \left\{ \begin{array}{ll}
Q \xrightarrow{C[-]} Q' \text{ and } P' \# B \# Q' & \text{if } C[-] \in L; \\
C[Q] \sim Q' \text{ and } P' \# B \# Q' & \text{otherwise.}
\end{array} \right.
\end{cases}
\]

The largest \( L \)-bisimulation is called \( L \)-bisimilarity and denoted by \( \sim_L \).

For instance, the relation \( B \) of Example 4.4 is an \( L_\ast \)-bisimulation for \( L_\ast \) being the set of contexts of the shape \( - \) or \( -|a.S + N \). Instead, for the set \( C \) of all contexts, \( B \) is not \( C \)-bisimilarity: the transition \( a.\overline{a} + \tau.0 \xrightarrow{a} A \) should be matched by a transition with the same label from \( \tau.0 \). More generally, \( a.\overline{a} + \tau.0 \) and \( \tau.0 \) are not \( C \)-bisimilar. The next
section shows that \( \sim_L \) coincides with barbed saturated bisimilarity for a suitable choice of \( L \). In the remainder of this section we prove that \( \sim_L \) is a congruence, as long as \( L \) verifies an additional condition.

**Definition 5.2.** A set of contexts \( L \subseteq C \) is closed under decomposition if whenever \( (C_1[-], C_2[-]) \) is a universal decomposition pair for \( C'[C[P]] \sim P' \) (see Definition 3.10) and \( C'[\cdot] \in L \) then \( C_1[-] \in L \).

Closure under decomposition guarantees that the first component of any universal decomposition pair reflects the fact of belonging to \( L \), so that it can be suitably used in the bisimulation game. In fact, the notion is sufficient for proving the result below.

**Proposition 5.3.** Let \( L \subseteq C \) a set of contexts. If a context LTS is decomposable and \( L \) is closed under decomposition, then \( \sim_L \) is a congruence.

**Proof.** In order to prove that \( \forall C[-], P \sim_L Q \) implies \( C[P] \sim_L C[Q] \), we show that

\[
\mathcal{B} = \{(C[P], C[Q]) \mid P \sim_L Q, C[-] \in C\}
\]

is a \( L \)-bisimulation.

Let us assume that \( C[P] \sim_L C[P'] \). By soundness, we know that \( C'[C[P]] \sim P' \). There are two cases: either \( P \) is active or it is inactive.

1. **If it is active then**, since the LTS is decomposable, there exists a decomposition pair \((C_1[-], C_2[-])\) such that, for all processes \( S \), whenever \( S \xrightarrow{C_1[-]} S'' \) then \( C[S] \xrightarrow{C[-]} S' \)

with \( S' = C_2[S''] \). There are two more cases.

   a. **If \( C[-] \in L \) then also \( C_1[-] \in L \), because \( L \) is closed under decomposition.** Since

   \( P \sim_L Q \), we have that \( Q \xrightarrow{C_1[-]} Q'' \) and \( P'' \sim_L Q'' \). Since \( Q \xrightarrow{C_1[-]} Q' \) and \( Q' = C_2[Q''] \), we have that

   \( C[Q] \xrightarrow{C[-]} C'[Q'] \) and \( Q' = C_2[Q''] = C_2[Q'] \). Since \( P'' \sim_L Q'' \), \( P' = C_2[P''] \) and \( C_2[Q''] = Q' \).

   b. **If \( C[-] \notin L \) then either \( C_1[-] \in L \) or \( C_1[-] \notin L \).** In both cases, from \( P \xrightarrow{C_1[-]} P'' \) we derive that \( C_1[Q] \sim Q'' \) and \( P'' \sim_L Q'' \), because \( Q \xrightarrow{C_1[-]} Q'' \) implies \( C_1[Q] \sim Q'' \).

   Since \( C_2[-] \) is reactive, we have that \( C_2[Q_1] \sim C_2[Q''] \), that is, \( C'[C[Q]] \sim C_2[Q''] \). Finally, since \( P'' \sim_L Q'' \), \( P' = C_2[P''] \) and \( C_2[Q''] = Q' \).

2. **If \( P \) is inactive in \( C'[C[P]] \sim P' \) with \( P' = D[P] \) then, since the LTS is decomposable, we have that \( C[Q] \xrightarrow{C[-]} Q' \), with \( Q' = D[Q] \).** So, since \( P \sim_L Q \), \( P' = D[P] \) and \( D[Q] = Q' \). This works for both the cases where \( C'[\cdot] \in L \) and \( C'[\cdot] \notin L \), because from \( C[Q] \xrightarrow{C[-]} Q' \) we conclude by soundness that \( C'[C[Q]] \sim Q' \).

\[ \square \]

Clearly \( \sim_L \) coincides with the standard bisimilarity \( \sim \) if \( L \) is the whole class of contexts. Since in this case \( L \) is closed under decomposition, Proposition 3.11 on the congruence for \( \sim \) on decomposable LTSs is an immediate consequence of the above result.

5.1.1. Barbed Saturated Bisimilarity via \( L \)-bisimilarity. Even if \( L \)-bisimilarity is a congruence, nothing ensures that it coincides with barbed saturated bisimilarity. Depending on the choice of \( L \) one obtains different equivalences: the smaller \( L \) is, the coarser \( \sim_L \). In \( \text{ACCS} \), for instance, for the set \( C \) of all contexts \( \sim_L \) is strictly included in \( \sim_{BS} \); when \( L \) is the set \( L_A \) defined above, then \( \sim_L = \sim_{BS} \). Instead, in \( \text{CCS} \) \( \sim_L \) coincides with \( \sim_{BS} \).

In this section we study those conditions that \( L \) should satisfy for ensuring that \( L \)-bisimilarity characterises barbed saturated bisimilarity.
In order to guarantee that \( \sim^{L} \subseteq \sim^{BS} \), we need some additional conditions intuitively ensuring that checking the barbs of \( \sim^{BS} \) is implicitly done in \( \sim^{L} \) by the labels in \( L \).

**Definition 5.4.** \([O\text{-}capture]\) A set \( L \subseteq C \) of contexts is \( O\text{-}capturing \) if for each barb \( o \in O \) there exists a context \( C[-] \in L \) such that for any process \( P \) we have \( P \downarrow o \) if and only if \( P \Downarrow C[-] \).

**Example 5.5.** Let us consider a set of CCS labels containing at least all the contexts of the shape \( -|a.S + N \) and \( -|\overline{a}.S + N \). It is easy to show that this set is \( O_{a} \)-capturing. Indeed, by definition \( P \downarrow a \) when \( P \equiv a.Q + M|R \), which is equivalent to say that \( P -|a.S+N|\sigma' \ Q|R|S \). In the same way, \( P \downarrow \bar{a} \) if and only if \( P -|\bar{a}.S+N|\sigma' \ Q|R|S \). Analogously, we can show that for the ACCS, a set of \( O_{A} \)-capturing contexts must contain at least the labels of the shape \( -|a.S + N \).

Next definition is useful to ensure that \( \sim^{BS} \subseteq \sim^{L} \).

**Definition 5.6 (Stable context).** Let \( R \subseteq P \times P \) be a relation and \( C[-] \in C \) a context. We say that \( C[-] \) is stable under \( R \) if for any two processes \( P \), \( Q \) whenever \( P R Q \) and \( P \Downarrow C[-] \) \( P' \) there exists \( Q' \) such that \( Q \Downarrow C[-] Q' \) and \( P'RQ' \). We say that \( C[-] \) is weakly stable under \( R \) if it satisfies the condition above when replacing \( \rightarrow \) by \( \Rightarrow \).

Stable contexts are discussed in depth in the next section. Here, we show the main correspondence results concerning \( L \)-bisimilarity.

**Theorem 5.7.** If the barbs in \( O \) are contextual, \( L \) is \( O\text{-}capturing \), and its contexts are stable under \( \sim^{BS} \) then \( \sim^{L} \) coincides with \( \sim^{BS} \).

**Proof.** In order to prove that \( \sim^{BS} \subseteq \sim^{L} \), we show that

\[
B = \{ (P,Q) \mid P \sim^{BS} Q \}
\]

is a \( L \)-bisimulation. Suppose that \( P \Downarrow C[-] \) \( P' \). We have two cases:

1. When \( C[-] \in L \), then \( C[-] \) is stable under \( \sim^{BS} \) and, since \( P \sim^{BS} Q \), we have \( Q \Downarrow C[-] Q' \) and \( P' \sim^{BS} Q' \).
2. When \( C[-] \notin L \), note that by soundness \( C[P] \sim Q' \). And since \( P \sim^{BS} Q \), \( C[Q] \sim Q' \) and \( P' \sim^{BS} Q' \).

Now, to prove that \( \sim^{L} \subseteq \sim^{BS} \), we show that

\[
B = \{ (P,Q) \mid P \sim^{L} Q \}
\]

is a barbed semi-saturated bisimulation: thus \( \sim^{L} \subseteq \sim^{BS} \) and the result follows from Proposition 4.2.

At first, we note that, since \( O \) is a set of contextual barbs, in order to show that \( B \) satisfies the first condition of Definition 4.1 it suffices to show that \( P \downarrow o \) implies \( Q \downarrow o \). Since \( L \) is \( O\text{-}capturing \), if \( P \downarrow o \) then there is a context \( C[-] \in L \) such that \( P \downarrow o \) if and only if \( P \Downarrow C[-] \). Since \( P \downarrow L \), we have \( Q \Downarrow C[-] \) and \( Q \downarrow o \).

In order to prove the second condition of Definition 4.1, it is enough to note that dealing with sound LTSs we have that \( P \Downarrow C[-] \) \( P' \) implies \( C[Q] \sim Q' \) with \( P' \sim^{L} Q' \).

---

5.1.2. Proving Context Stability for the Running Examples. The key step for characterising \( \sim^{BS} \) via \( L \)-bisimilarity is the identification of those contexts which are stable under \( \sim^{BS} \). The approach used for CCS and ACCS that is illustrated below consists in showing
for each context $C[-]$ a binary predicate $\mathcal{P}^{C[-]}(X,Y) \subseteq \mathcal{P} \times \mathcal{P}$ defined by means of barbs, contexts and reductions such that $\mathcal{P}^{C[-]}(X,Y)$ if and only if $X \sim_{\mathcal{C}C\mathcal{S}} Y$.

For CCS, the contexts having one of the shapes

$$-a.S + N, -a.S + N,$$

for $S$ and $N$ ranging over the set of CCS processes and summations, respectively, are stable under $\sim_{BS}$. This is proved by using the predicates in Fig. 4. Since these are defined just with barbs, contexts and reductions, it is easy to prove the following.

**PROPOSITION 5.8.** Let $C[-]$ be a CCS context whose shape is in $\circledS$. If $P \sim_{BS} Q$ and $\mathcal{P}^{C[-]}(P,P')$ (see Fig. 4) then there exists $Q'$ such that $\mathcal{P}^{C[-]}(Q,Q')$ and $P' \sim_{BS} Q'$.

**PROOF.** We show the proof just for $C[-] = -a.S + N$: the other cases are analogous. Assume that $P \sim_{BS} Q$ and that $\mathcal{P}^{C[-]}(P,P')$ holds. This means that $P|a.(i)S|i \sim P'$, and since $P \sim_{BS} Q$, also $Q|a.(i)S|i \sim Q''$ with $P'' \sim_{BS} Q''$. Therefore, since $P'' \Downarrow$, also $Q'' \Downarrow$. Moreover, we know $P'' \sim P' \not\Downarrow$, thus also $Q'' \sim Q'$ with $P' \sim_{BS} Q'$. This means that also $Q' \Downarrow$, and so $\mathcal{P}^{C[-]}(Q,Q')$ holds. □

**PROPOSITION 5.9.** Let $C[-]$ be a CCS context whose shape is in $\circledS$ and $\mathcal{P}^{C[-]}(X,Y)$ as defined in Fig. 4. Then $X \sim_{\mathcal{C}C\mathcal{S}} Y$ if and only if $\mathcal{P}^{C[-]}(X,Y)$.

**PROOF.** We prove the proposition for the three types of contexts.

— If $X \not\Downarrow a.S + N_{\sigma}$, $Y$, then $X \equiv a.Q + M|R$ and $Y \equiv Q|R|S$. We have that $X|a.(i)S|i \sim P' = Q|R|S|i$ and $P' \Downarrow$. Moreover $P' \sim Y = Q|R|S$ and $Y \not\Downarrow$, because $i \notin fn(X) \cup fn(S)$.

Reciprocally, when $P \not\Downarrow a.S + N(X,Y)$ holds we have (a) $X|a.(i)S|i \sim P' \Downarrow$ and (b) $P' \sim Y \not\Downarrow$. Since $i \notin fn(X)$, (a) entails $X \equiv a.Q + M|R$ and $P' \equiv Q|R|S|i$ and (b) entails $Y \equiv Q|R|S$. By definition of $\sigma'$, $X \sim_{-a.S + N_{\sigma'}} Q|R|S \equiv Y$.

— If $X \not\Downarrow a.S + N_{\sigma'}$, $Y$, we proceed as above by replacing $\bar{a}$ with $a$.

The case of identity context is immediate.

□

Now, let us consider $L_{\sigma'}$, the set of contexts given by the labels of the LTS $\sigma'$. The previous propositions ensure that all these contexts are stable under $\sim_{BS}$ and moreover, as discussed in Section 5.1.1, they capture the synchronous barbs $O_s$. Therefore, thanks to Theorem 5.7, $L_{\sigma'}$-bisimilarity coincides with $\sim_{BS}$ for CCS.

It is worth to note that $L_{\sigma'}$-bisimilarity is just the ordinary notion of bisimilarity on the LTS $\sigma'$. Note also the close correspondence between $\sigma'$ and the standard LTS for CCS [Milner 1989]: the context $-$ corresponds to the label $\tau$, $-a.S + N$ to $\bar{a}$ and

---

2That is, precisely those contexts that may occur as labels in $\sigma'$, up-to congruence
A General Theory of Barbs, Contexts and Labels

For \textsc{ACCS}, instead, only the contexts having one of the shapes

\[ -|a.S + N, - \]

for \( S \) and \( N \) ranging over the set of \textsc{ACCS} processes and summations, respectively, are stable under \( \sim^{BS} \). This can be proved exactly in the same way as for the synchronous case, by using the predicates in Fig. 4, except for the first one, \( \mathcal{P} \sim a.S + N(X, Y) \). The predicate cannot be defined in \( \textsc{ACCS} \), since there exists no context \( -|\bar{a}([i], S)|i \). For the same reason, it is impossible to define a similar predicate for the contexts \( -|\bar{a} \), which, indeed, are not stable under \( \sim^{BS} \).

By taking \( L_\alpha \) to be the set contexts as in \( \bar{a} \), we have that \( L_\alpha \) captures the asynchronous barbs \( O_\alpha \) and therefore, by Theorem 5.7, \( L_\alpha \)-bisimilarity coincides with \( \sim^{BS} \) for \textsc{ACCS}. It is worth to observe that \( L_\alpha \)-bisimilarity closely corresponds to asynchronous bisimilarity [Amadio et al. 1998]. In Appendix B, we report the original definition from [Amadio et al. 1998] and we prove a formal correspondence between the two equivalences. In the remainder of this section, we give an intuitive explanation.

As for the synchronous case, there is a strong correspondence between the transitions of the context LTS \( \alpha' \) and the transition of the standard LTS (reported in Appendix B): the context \( \sim \) corresponds to the label \( \tau \), \( -|a.S + N \) corresponds to \( \bar{a} \) and \( -|\bar{a} \) corresponds to \( a \). In \( L_\alpha \)-bisimilarity, the transitions of a process \( P \) labeled with \( \bar{a} \) or \( -|a.S + N \) must be matched by those of an equivalent process \( Q \) that are labeled in the same way. Instead, the transitions labeled with \( -|\bar{a} \) can be matched by reductions of \( Q|\bar{a} \). For instance, the transition \( a.\bar{a} + \tau.0 \sim a.\bar{a} \) \( a.\bar{a} + \tau.0 \rightarrow a \), but \( \tau.0 \not\rightarrow a.\bar{a} \).

This can be proved exactly in the same way as for the synchronous case.

5.2. Weak case

The weak counterpart of \( L \)-bisimilarity is defined as the strong one (Definition 5.1).

**Definition 5.10 (Weak \( L \)-Bisimilarity).** Let \( L \subseteq C \) be a set of contexts. A symmetric relation \( B \subseteq \mathcal{P} \times \mathcal{P} \) is a weak \( L \)-bisimulation if whenever \( P \triangleright B \triangleright Q \) then

if \( P \triangleright C \triangleright P' \) then

\[
\begin{align*}
Q \triangleright C \triangleright Q' \quad & \text{and} \quad P' \triangleright B \triangleright Q', \quad \text{if} \ C \triangleright \in L; \\
C \triangleright \triangleright Q' \quad & \text{and} \quad P' \triangleright B \triangleright Q', \quad \text{otherwise}.
\end{align*}
\]

The largest weak \( L \)-bisimulation is called weak \( L \)-bisimilarity and denoted by \( \approx^L \).

As for the strong case, we show that \( \approx^L \) is a congruence, under suitable conditions. In this case, to prove that \( \approx^L \) is a congruence we need an additional condition: all the contexts in \( C \) have to be reactive (Definition 3.3).

**Proposition 5.11.** Let \( L \subseteq C \) a set of contexts. If a context LTS is decomposable, the contexts in \( C \) are reactive and \( L \) is closed under decomposition, then \( \approx^L \) is a congruence.
\[\mathcal{P}^{-|\bar{a}.S + N}(X, Y) \quad \exists i \notin fn(X) \cup fn(S) \text{ s.t. } X|\bar{a}.(i|S)|i \leadsto^* Y \not\Downarrow_i\]
\[\mathcal{P}^{-a.S + N}(X, Y) \quad \exists i \notin fn(X) \cup fn(S) \text{ s.t. } X|a.(i|S)|i \leadsto^* Y \not\Downarrow_i\]

**Fig. 5. Predicates for CCS and ACCS**

**Proof.** We can repeat the same proof of Proposition 5.3 by replacing \(\sim^L\) with \(\approx^L\).

At point (1)(a), one deduces that \(Q \not\Downarrow^{C[\bar{C}]} Q''\) (namely \(Q \leadsto^* Q_0 \not\Downarrow^{C[\bar{C}]} Q_1 \leadsto^* Q''\)) and \(Q_0 \not\Downarrow^{C[\bar{C}]} Q'\). In order to conclude that \(Q \not\Downarrow^{C[\bar{C}]} Q''\) (namely \(Q \leadsto^* Q_0 \not\Downarrow^{C[\bar{C}]} C_2[Q'']\)) we need \(C[-]\) and \(C_2[-]\) to be reactive.

Similarly, at point (1)(b), to conclude \(C_1[Q] \leadsto^* Q''\) from \(Q \not\Downarrow^{C_1[\bar{C}]} Q''\), we need \(C_1[-]\) to be reactive. \(\square\)

The weak equivalence obtained by taking \(L\) to be the set of all contexts \((\approx)\) is not interesting per se, since it does not coincide with the standard notion of weak bisimilarity [Milner 1989]. Indeed, in \(\approx\) a transition labeled with \(\tau\) (intuitively corresponding to a \(\tau\) transition) must be matched by \(\Rightarrow\) (intuitively corresponding to one or more \(\tau\) transitions), while in weak bisimilarity a \(\tau\) transition can be matched also by an empty sequence of \(\tau\) transitions. We come back on this point later on, in Section 5.2.2.

**5.2.1. Weak Barbed Saturated Bisimilarity via Weak L-bisimilarity.** We now provide sufficient conditions ensuring that weak \(L\)-bisimilarity characterises weak barbed saturated bisimilarity. The only difference with the strong case is that contexts must be weakly stable (Definition 5.6) and that the contexts in \(L\) must be reactive.

**Theorem 5.12.** If the bars in \(O\) are weak contextual, \(L\) is \(O\)-capturing and its contexts are reactive and weakly stable under \(\approx^L\), then \(\approx^L = \approx^L\).

**Proof.** The proof is completely analogous to the one of Theorem 5.7. The additional hypothesis that all contexts are reactive is needed in the second part of the proof (namely to show that \(\approx^L \subseteq \approx^L\)) for the second condition of Definition 4.5: one deduces that \(Q \not\Downarrow^{C[-]} Q'\) and needs \(C[-]\) to be reactive in order to conclude that \(C[Q] \leadsto^* Q'\). \(\square\)

**5.2.2. Proving Context Weak Stability for the Running Examples.** In order to identify those contexts which are weakly stable under \(\approx^L\), we use binary predicates defined by means of weak bars, contexts and \(\leadsto^*\).

For CCS, the contexts having one of the shapes
\[-|\bar{a}.S + N, -|a.S + N\]
are weakly stable under \(\approx^L\). This is proved in a similar way to the strong case (Section 5.1.2) by using the predicates in Fig. 5. Analogously, one can prove for ACCS that the contexts having the shape
\[-|a.S + N\]
are weakly stable under \(\approx^L\). Since these two sets of contexts capture the bars (the first one for CCS and the second one for ACCS) and since all contexts in \(R'_L\) and in \(R'_L\) are reactive, one can apply Theorem 5.12 to obtain a characterization of \(\approx^L\) for CCS and ACCS by means of \(L\)-bisimilarity.

It is worth to note that for both CCS and ACCS the context \(-\) is not weakly stable under \(\approx^L\). This is the case for most process calculi: for \(P \approx^L Q\) and \(P \not\rightarrow P'\) (i.e., \(P \not\approx P'\)), it is not guaranteed that \(Q \not\rightarrow\). In terms of \(L\)-bisimilarity, this means that...
In order to introduce them, we need MAs contexts in [Rathke and Sobociński 2008] and [Merro and Zappa Nardelli 2005], respectively, and strong and weak barbs to define the standard equivalences for MAs: strong and weak reduction barbed congruence, of which a labelled characterisation is in [Rathke and Sobociński 2008] and [Merro and Zappa Nardelli 2005], respectively. In order to introduce them, we need MAs contexts: they are MAs processes with a hole.

6. MOBILE AMBIENTS

This section presents an application of our framework to the calculus of Mobile Ambients (MA). First, we recall its syntax and semantics (Section 6.1) and then we show how the latter can be checked by means of barbed semi-saturated and L-bisimulations.

6.1. Syntax and semantics

This section reports the finite, communication-free fragment of MAs: its syntax, reduction semantics and standard behavioural equivalences [Cardelli and Gordon 2000].

The syntax is shown in Fig. 6(a). We assume a set $N$ of names ranged over by $m,n,o,\ldots$ and we let $P,Q,R,\ldots$ range over the set $P_M$ of processes. The free names of a process $P$ (denoted by $fn(P)$) are defined as usual. Processes are taken up to a structural congruence, axiomatised in Fig. 6(b) and denoted by $\equiv$. The reduction relation $\leadsto_M$ describes process evolution: it is the least relation $\leadsto_M$: $P_M\times P_M$ closed under $\equiv$ and generated by the rules in Fig. 6(c).

As said in Section 2.2, a barb $o$ is a predicate over processes, with $P \downarrow o$ denoting that $P$ satisfies $o$. In MAs, $P \downarrow n$ denotes the presence at top-level of an unrestricted ambient $n$. Formally, $P \downarrow n$ if $P = (\nu A)(n)[Q][R]$ and $n \notin A$, for some processes $Q$ and $R$ and a set of restricted names $A$. A process $P$ satisfies the weak barb $n$ (denoted as $P \downarrow n$) if there exists a process $P'$ such that $P \leadsto_M P'$ and $P' \downarrow n$.

Strong and weak barbs are exploited to define the standard equivalences for MAs: strong and weak reduction barbed congruence, of which a labelled characterisation is in [Rathke and Sobociński 2008] and [Merro and Zappa Nardelli 2005], respectively.

Fig. 6. (a) Syntax, (b) structural congruence and (c) reduction relation of MAs.

whenever $L$ contains only weakly stable contexts, if $P \rightarrow P'$ then $Q \rightarrow^* Q'$, that is, either $Q$ makes one or more reductions, or it makes no reductions. This is analogous to the standard notion of weak bisimilarity for CCS [Milner 1989]: when $P$ makes a $\tau$ transition, $Q$ can answer with either 0 or more $\tau$ transitions, while when $P$ performs transitions labeled with $a$ or $\bar{a}$ (corresponding to $-|a.S + N$ and $-|\bar{a}.S + N$), $Q$ must answer with (weak) transitions with the same label.

For ACCS, one can use similar arguments to those of the strong case to show that weak $L$-bisimilarity coincides with the weak asynchronous bisimilarity introduced in [Amadio et al. 1998]. See Appendix B for more details.
−, formally generated by the following grammar (for $R \in \mathcal{P}_\mathbb{M}$)
\[
C[-] ::= -, n[C[-]], M.C[-], (\nu n)C[-], C[-] \mid R
\]
and, as for CCS and ACCS, taken up to congruence (assuming $fn(\cdot) = N$).

**Definition 6.1 (Reduction Barbed Congruences).** Strong reduction barbed congruence $\sim_{\mathbb{M}}$ is the largest symmetric relation $\mathcal{B}$ such that whenever $P \mathcal{B} Q$ then

1. if $P \Downarrow n$ then $Q \Downarrow n$;
2. if $P \sim_{\mathbb{M}} P'$ then $Q \sim_{\mathbb{M}} Q'$ and $P' \mathcal{B} Q'$;
3. $\forall C[-].C[P] \mathcal{B} C[Q]$.

Weak reduction barbed congruence $\approx_{\mathbb{M}}$ is the largest symmetric relation $\mathcal{B}$ such that whenever $P \mathcal{B} Q$ then

1. if $P \Downarrow n$ then $Q \Downarrow n$;
2. if $P \sim_{\mathbb{M}} P'$ then $Q \sim_{\mathbb{M}} Q'$ and $P' \mathcal{B} Q'$;
3. $\forall C[-].C[P] \mathcal{B} C[Q]$.

It is easy to see that the above notion is closely related to barbed saturated semantics (Definition 2.4) and, indeed, as we show in Section 6.3, $\sim_{\mathbb{M}}$ and $\approx_{\mathbb{M}}$ are special instances of $\sim^{\mathcal{B}S}$ and $\approx^{\mathcal{B}S}$, in a given reactive system $\mathbb{M}$. This fact, together with the LTSs that we introduce in the next section, allows to use semi-saturated and $L$-bisimulations for proving $\sim_{\mathbb{M}}$ and $\approx_{\mathbb{M}}$. First, we provide some examples for $\sim_{\mathbb{M}}$ and $\approx_{\mathbb{M}}$.

**Example 6.2.** It is easy to see that $(\nu n)(n[0]) \sim_{\mathbb{M}} 0$: none of the two processes expose a barb, perform a reduction or, since the ambient $n$ is restricted, can interact with a context. By condition (3) of Definition 6.1, $\sim_{\mathbb{M}}$ is a congruence and therefore, for any ambient name $m \in \mathcal{N}$ and process $S \in \mathcal{P}_{\mathbb{M}}$, $m[(\nu n)(n[0])][S] \sim_{\mathbb{M}} m[0][S]$. From the structural congruence $\equiv$ we have that if $m \neq n$ and $n \notin fn(S)$, $(\nu n)(m[n[0][S]]) \sim_{\mathbb{M}} m[S]$. The latter equivalence will be useful later to prove that

\[
P = (\nu n)(n[in.m.0]) \quad \text{and} \quad Q = 0
\]
are in $\approx_{\mathbb{M}}$. This is an instance of a well-known equivalence of MAs, called the perfect firewall equation [Cardelli and Gordon 2000].

### 6.2. A labeled transition system for Mobile Ambients

In Fig. 7 we present the LTS $\mu$ for MAs proposed in [Bonchi et al. 2009c]. Informally, the labels on transitions are the “minimal” contexts allowing a reduction to occur. Recall that processes and labels of the LTS are closed with respect to structural congruence.

The rule TAU models the stand-alone reductions of a process. On the contrary, the rule EXTRED models the reduction performed by the context without any interaction with the process (i.e., the process is inactive). More precisely, EXTRED represents three rules, one for each axiom of the reduction relation (see Fig. 6(c), left): the process $L$ of the label represents the left hand side of the axiom and the process $R$ of the target state represents the right hand side.

The other rules model the interactions of a process with its environment. Note that in the conclusion of some rules there are names and processes (denoted by $o$, $S_1$, $S_2$) that do not appear in the premises. These represent ambient names and processes that are provided by the environment and thus we must always assume that $\{o\} \cup fn(S_1) \cup fn(S_2) \cap A = \emptyset$. Intuitively, the rule OPEN enables a process to open a new ambient provided by the environment, while the rule COOPEN allows the environment to open a sibling ambient of the process. The rule INAMB enables an ambient of the process to migrate into a sibling ambient provided by the environment, while in the rule IN both the ambients are provided by the environment. In the rule COIN an ambient provided
A General Theory of Barbs, Contexts and Labels

(Tau) \[ P \rightsquigarrow_m Q \]

(ExtRed) \( P \rightsquigarrow_m Q \)

(Out) \( P \equiv (\nu A)(\text{out } m.P_1|P_2) \quad m \not\in A \)

by the environment enters an ambient of the process. The rule OUTAMB models an
ambient of the process exiting from an ambient provided by the environment, while in
the rule OUT both ambients are provided by the environment.

Example 6.3. The MAs process \( P \) of Example 6.2 performs the transition \( P \rightsquigarrow_m (\nu n)(m|n|0|S) \) thanks to the rule INAMB. Apart from this, \( P \) can only perform transitions generated by the rule EXTRED.

6.3. A reactive system for Mobile Ambients

This section proposes two labelled characterisations of (weak) reduction barbed con-
gruences for MAs by means of the semi-saturated bisimilarity and \( L \)-bisimilarity.

First of all, we show that MAs fits in the theory of Section 2. We consider the system
theory \( \mathcal{S}_M = \langle \mathcal{P}_M, \mathcal{C}_M, \cdot \rangle \), where \( \mathcal{C}_M \) is the monoid \( \langle \mathcal{C}_M, \circ, - \rangle \), with \( \mathcal{C}_M \) the set of unary MAs contexts presented in \( C \). The calculus can thus be seen as a reactive system \( \mathcal{R}_M = \langle \mathcal{S}_M, \rightsquigarrow_m, O_M \rangle \), where \( O_M \) is the set of MAs barbs introduced in Section 6.1.

Proposition 6.4. \( \rightsquigarrow_m = \approx_{BS}^{R_M} \) and \( \approx_{BS}^{R_M} \).

Proof. Immediate. \( \square \)

Now, we would like to exploit the LTS \( \mu \) introduced in the previous section for proving \( \rightsquigarrow_m \) and \( \approx_{BS}^{R_M} \) by means of (weak) barbed semi-saturated bisimilarity and (weak) \( L \)-bisimilarity, but first, we should show that \( \mu \) is a sound and actively complete con-
text LTS. Instead of proving it directly, we consider a simpler reactive system, which
differs from \( \mathcal{R}_M \) only with respect to the monoid of contexts: \( \mathcal{C}'_M \) contains the unary MAs contexts generated by the following grammar (for \( R \in \mathcal{P}_M \))

\[
C[-] ::= -, n[C[-]] \quad (\nu n)C[-], \quad C[-] \mid R
\]

We consider the monoid of contexts \( \mathcal{C}'_M = \langle \mathcal{C}'_M, \circ, - \rangle \), the system theory \( \mathcal{S}'_M = \langle \mathcal{P}_M, \mathcal{C}'_M, \cdot \rangle \) and the reactive system \( \mathcal{R}'_M = \langle \mathcal{S}'_M, \rightsquigarrow_m, O_M \rangle \). Since the contexts in \( \mathcal{C}_M \) and not in \( \mathcal{C}'_M \) are of the shape \( M.C[-] \) and these are non-discriminating, we can exploit Proposition 2.11.
Proposition 6.5. \( \sim_M = \sim_{BSS}^{\mu} \) and \( \approx_M = \approx_{BSS}^{\mu} \).

It is easy to check that \( \mu \) is sound and complete.\(^3\) Again, as for CCS and ACCS, the transitions generated by the rule \textsc{ExtRed} are inactive (Definition 3.7). We remove them from \( \mu \) and we obtain a new LTS \( \mu' \) that is sound and actively complete for \( \mathbb{R}_M' \) by Proposition 3.8.

As shown in Section 4, we can efficiently characterise strong and weak barbed semantics by exploiting the semi-saturated game. In order to characterise \( \sim_M \) and \( \approx_M \), we apply respectively Definitions 4.1 and 4.5 to the LTS \( \mu' \). By the way, the arbitrary contexts \( C[-] \) in condition (1) of these definitions can be fixed to the identity \( - \) since, as stated by the proposition below, barbs are contextual.

**PROPOSITION 6.6.** The barbs in \( O_M \) are both contextual and weak contextual.

**PROOF.** The proof is a routine check by induction on the structure of contexts. \( \square \)

**THEOREM 6.7.** \( \sim_M = \sim_{BSS}^{\mu} \) and \( \approx_M = \approx_{BSS}^{\mu} \).

**PROOF.** The two statements are instances of Propositions 4.2 and 4.6, respectively. \( \square \)

**Example 6.8.** The theorem above allows to prove that processes \( P \) and \( Q \) of Example 6.2 are in \( \approx_M \). First we recall that \( (\nu m)(m[n][0]|S]) \sim_M m[S] \) and thus, since \( \sim_M \subseteq \approx_M \), \( (\nu m)(m[n][0]|S]) \approx_M m[S] \). Then, we take \( B = \{(P, Q), (Q, P)\} \cup \approx_M \) and we prove that it is a semi-saturated bisimulation. We only check the pairs \( (P, Q) \) and \( (Q, P) \) since those in \( \approx_M \) are trivial, and

1. neither \( P \) nor \( Q \) satisfy any barb;
2. the transition \( P \vdash \langle m[S] \rangle \sigma' \) \((\nu m)(m[n][0]|S]) \) is matched by \( Q[m[S]] \sim^* Q[m[S]] \) and, since \( Q[m[S]] \equiv m[S] \), the target processes \( (\nu m)(m[n][0]|S]) \) and \( Q[m[S]] \) are in \( B \).

Note the similarities with Example 4.8 for ACCS: a labeled transition can be matched by an empty sequence of reductions.

The previous examples show that for MAs, similarly to ACCS and differently from CCS, not all the contexts occurring as labels in the LTS \( \mu' \) are stable (Definition 5.6). Contexts of the form \( -m[S] \) are not stable under \( \approx_{BSS} \), since \( P \approx_{BSS} Q \) and \( P \) performs a transition with such context while \( Q \) cannot. Therefore, in order to apply the results of Section 5, we need to identify a set of contexts \( L_M \) which captures the barbs in \( O_M \) and contains only stable contexts (as required by Theorems 5.7 and 5.12).

**PROPOSITION 6.9.** Let \( L_M \) be the set of contexts having the shape \(-\{\text{open } n.S\}, for n \in \mathbb{N} and S \in P_M. Then L_M is O_M-capturing.

**PROOF.** Given a barb \( n \in O_M \) and a process \( P \), it is immediate to see that \( P \downarrow_n \) if and only if \( P \vdash \langle \text{open } n.S \rangle \sim^*_\mu' P' \), for some process \( S \). \( \square \)

**PROPOSITION 6.10.** The contexts in \( L_M \) are reactive, stable under \( \sim_{BSS}^{\mu'} \) and weakly stable under \( \approx_{BSS} \).

\(^3\)As it occurred for CCS and ACCS, besides being easily proved, it is not a surprise: the LTS \( \mu \) has been derived by using the theory in [Sassone and Sobociński 2005], which is in turn a refinement of Leifer and Milner’s one. This fact ensures that both Property 1 and Property 2 of Definition 3.6 are satisfied.
\[ P \rightarrow \text{open} \ n.S(X,Y) \quad \exists \ P'' \text{ and } m \notin f(n(X) \text{ s.t. } P'' \downarrow_m, C'(X) \sim \mu P'' \sim \mu Y \text{ and } Y \not\downarrow_m \text{ with } C'[-] = - | \text{open} m(0) | \text{open} m.S \]

Fig. 8. Predicate for MAs.

**Proof.** It is immediate to see that the contexts in \( L_\mu \) are reactive. The check for stability and weak stability can be done exactly in the same way as for CCS and ACCS, but by using the predicate in Fig. 8. \( \square \)

**Corollary 6.11.** \( \sim_{BS}^{\mu} = \sim_{L_\mu}^{\mu} \) and \( \approx_{BS}^{\mu} = \approx_{L_\mu}^{\mu} \).

**Proof.** It follows from the previous two propositions via Theorems 5.7 and 5.12. \( \square \)

Differently from CCS, it is not possible to have a characterization of \( \sim_{BS}^{\mu} \) in terms of standard bisimulation via Theorem 5.7. Indeed, as it occurs for ACCS, not all the contexts appearing as labels in \( \mu' \) are (weakly) stable.

**7. Conclusions and Further Works**

The work presented in this paper stems from the research on reactive systems. The key idea underlying the theory by Leifer and Milner is to distill from an unlabeled reduction systems a labelled one by using as labels those contexts that are somehow minimal, thus guaranteeing that the associated bisimilarity is a congruence. From the theory a stream of results derived, often extending the class of systems under consideration or tackling e.g. barbed and weak bisimilarities.

This paper follows an orthogonal approach. Instead of a descriptive solution, identifying precisely the class of minimal contexts, our prescriptive proposal aims at characterizing those LTSs whose labels are contexts such that (1) they capture the barbed bisimilarities of the underlying reactive systems, and (2) such bisimilarities are congruences. Furthermore, a special emphasis is put in developing proof techniques that may ease the burden of proving these properties.

The key notions for ensuring the good behavior of a context LTS are soundness and active completeness. Even if we did not prove it, an LTS distilled by using minimal contexts is sound and complete, thus guaranteeing that our framework encompasses Leifer and Milner’s theory. In fact, the LTS presented in our MAs case study has been distilled using the borrowed contexts technique [Bonchi et al. 2009c]. We do believe that many more examples of contextual LTSs are just waiting to be further explored, as the recently proposal for concurrent constraint programming [Aristizabal et al. 2011] seems to suggest. To a certain extent, our proposal relies on the ingenuity of the researchers, offering a simple, set-theoretical framework where the good behavior of an “intuitively correct” LTS can be safely and easily verified.

We foresee three possible extensions.

First, note that this work basically considers untyped process calculi. We strove for a simple presentation of the framework, in order to highlight its intuitive appeal, and since most applications actually involve such calculi. However, a system theory can be safely extended in order to deal with multi-sorted calculi. A more abstract presentation would be obtained by considering a category whose objects are types and arrows are contexts, and taking a presheaf over it for the monoid action. Indeed, this is a rephrasing of the original definition of reactive systems by Leifer and Milner. The whole theory for sound and (actively) complete LTSs should work as smoothly using the more general formalism.
Second, we would like to address open systems, where the counterpart of a monoid of contexts is a monoid of substitutions. The topic has been tentatively addressed in categorical terms [Klin et al. 2005; Gadducci and Monreale 2013], yet the complexity of the machinery is further increased. We do believe that the categorical properties required to hold in these systems can be more easily recast in terms of suitable distributivity laws between the two monoid operators.

Finally, and perhaps mostly relevant for our theory, it would be nice to have a direct characterization of (broad families of) the systems that satisfy the properties of soundness and active completeness. This should be obtained by looking at the form of the rules defining the operational semantics and at the allowed observations, perhaps in a similar way as it happens for many SOS-like rule formats.

REFERENCES


A General Theory of Barbs, Contexts and Labels

A. ON BARBED BISIMILARITY

Barbs were originally introduced in [Milner and Sangiorgi 1992] as a basic concept for defining behavioural equivalences for those formalisms whose operational semantics is given by a reduction relation, rather than by an LTS. The original equivalence, namely barbed congruence, is defined as follows.

Definition A.1 (Barbed Bisimilarity, Barbed Congruence). A symmetric relation $\mathcal{R}$ is a barbed bisimulation if whenever $P \mathcal{R} Q$ then

- if $P \Downarrow o$ then $Q \Downarrow o$;
- if $P \Rightarrow P'$ then $Q \Rightarrow Q'$ and $P' \mathcal{R} Q'$.

Barbed bisimilarity $\sim^B$ is the largest barbed bisimulation; barbed congruence $\simeq^B$ is the largest congruence contained in $\sim^B$.

Barbed congruence is clearly a congruence, but there is no guarantee that it is also a bisimulation. Barbed saturated bisimilarity instead is both a congruence and a barbed bisimulation: actually it is the largest barbed bisimulation that is also a congruence. It is worth noting that $\sim^B_{BS}$ is finer than $\sim^B$ (the largest congruence contained into barbed bisimilarity). Intuitively, in the former case the external observer can plug processes into contexts at any step of their execution, while in the latter the observer can contextualize systems only at the beginning. The former observer is more powerful than the latter; thus proving that $\sim^B_{BS}$ is indeed finer than $\simeq^B$.

In our opinion $\sim^B_{BS}$ is more appropriate for modelling concurrent interactive systems embedded in an environment that continuously changes. And while the two notions often coincide [Fournet and Gonthier 1998], as we see in Section 6 for Mobile Ambients, the standard behavioural equivalence is an instance of $\sim^B_{BS}$.

For the sake of completeness, we recall that the notion of saturated bisimilarity (but without barbs) for reactive systems was originally proposed in [Bonchi and Montanari 2009]. Moreover, it is connected to asynchronous bisimilarity (that we are going to discuss below) was already hinted in [Bonchi and Montanari 2008].

B. ON ASYNCHRONOUS BISIMILARITY

This appendix discusses the correspondence between $\mathcal{L}_\Lambda$-bisimilarity and asynchronous bisimilarity [Amadio et al. 1998]. First, we report its original definition, using the LTS in Fig. 9 where we use $\mu$ to range over the set of labels $\{\tau, a, \bar{a} \mid a \in \mathcal{N}\}$.

---


ACM Transactions on Computational Logic, Vol. V, No. N, Article A, Publication date: January YYYY.
Definition B.1 (Asynchronous Bisimulation). A symmetric relation $R$ is an asynchronous bisimulation if whenever $P \not\sim R Q$ then

- if $P \xrightarrow{a} P'$ then $Q \not\rightarrow Q'$ and $P' R Q'$,
- if $P \xrightarrow{\tau} P'$ then $Q \not\rightarrow Q'$ and $P' R Q'$,
- if $P \xrightarrow{\tau} P'$ then either $Q \not\rightarrow Q'$ or $P' R Q'$ or $Q \not\rightarrow Q'$ and $P' R Q'|a$.

Asynchronous bisimilarity $\sim^A$ is the largest asynchronous bisimulation.

Before stating our result, it is important to observe that there is a close correspondence between the standard LTS (Fig. 9) and the context LTS $\alpha'$:

$$
P \xrightarrow{a} Q \text{ iff } P \xrightarrow{\alpha a'} Q
\quad
P \xrightarrow{\bar{a}} Q \text{ iff } P \xrightarrow{\bar{a} \cdot} Q
\quad
P \xrightarrow{\tau} Q \text{ iff } P \xrightarrow{\tau} Q_S.$$

**Proposition B.2.** $\sim^A = \sim^L_{\alpha'}$

**Proof.** In order to prove that $\sim^L_{\alpha'} \subseteq \sim^A$, we show that

$$
B = \{(P, Q) \mid P \not\sim^A Q\}
$$
is an asynchronous bisimulation. Suppose that $P \not\sim^L_{\alpha'} P'$, then $\mu$ is either an input, an output or a $\tau$. If it is an output or a $\tau$, the proof is trivial the correspondence above (recall that $L_{\alpha'}$ is the set of contexts of the shape $-|a.S + N, \cdot$). If it is an input, then $P \not\sim^L_{\alpha'} P'$. Since $P \sim^L_{\alpha'} Q$ and $-|a \not\in L_{\alpha'}$, we have that $Q|a \not\sim Q'$ and $P' \not\sim^L_{\alpha'} Q'$. This is equivalent to require that $Q \not\rightarrow Q'$ and $P' \not\sim^L_{\alpha'} Q'$ or $Q \sim Q''$ and $P' \sim^L_{\alpha'} Q'' = Q''|a$. This means that either $Q \not\rightarrow Q'$ and $P' B Q'$ or $Q \not\rightarrow Q''$ and $P' B Q''|a$.

In order to prove that $\sim^A \subseteq \sim^L_{\alpha'}$, we show that

$$
B = \{(P, Q) \mid P \sim^A Q\}
$$
is an L-bisimulation. Suppose that $P \not\sim^L_{\alpha'} P'$. Again we have three cases and the only non trivial is for $C[\cdot] = -|\bar{a}$ (corresponding to an input). Suppose that $P \not\sim^L_{\alpha'} P'$. This means that $P \not\rightarrow P'$ and, since $P \sim^A Q$, either $Q \not\rightarrow Q'$ and $P' \sim^A Q'$ or $Q \sim Q''$ and $P' \sim^A Q'' = Q''|\bar{a}$. This corresponds to say that $Q|a \not\sim Q'$ and $P' B Q'$. □

Weak asynchronous bisimilarity [Amadio et al. 1998] saturates the LTS in Fig. 9, where $P \xrightarrow{\tau} P'$ stands for $P \sim^* P'$ and $P \xrightarrow{\mu} P'$ for $P \sim^* \bullet \xrightarrow{\mu} \bullet \sim^* P'$ with $\mu \not\equiv \tau$.

**Definition B.3 (Weak Asynchronous Bisimilarity).** Weak asynchronous bisimilarity $\approx^A$ is the largest symmetric relation $B$ such that $P B Q$ implies

- if $P \xrightarrow{a} P'$ then $Q \xrightarrow{\alpha a} Q'$ and $P' B Q'$,
- if $P \xrightarrow{\tau} P'$ then $Q \xrightarrow{\tau} Q'$ and $P' B Q'$,
- if $P \xrightarrow{\tau} P'$ then either $Q \xrightarrow{\tau} Q'$ and $P' B Q'$ or $Q \xrightarrow{\tau} Q'$ and $P' B Q'|\bar{a}$.

By taking $L_{\alpha'}$ to be the set of contexts of the shape $-|a.S + N$, one can prove that weak $L_{\alpha'}$-bisimilarity coincides with $\approx^A$.

**Proposition B.4.** $\approx^A = \sim^L_{\alpha'}$.

**Proof.** The proof is analogous to the one of the strong case. □