MODELLING MAC-LAYER COMMUNICATIONS IN WIRELESS SYSTEMS

ANDREA CERONE, MATTHEW HENNESSY, AND MASSIMO MERRO

IMDEA Software Institute, Spain
e-mail address: andrea.cerone@imdea.org

School of Statistics and Computer Science, Trinity College Dublin, Ireland
e-mail address: Matthew.Hennessy@cs.tcd.ie

Dipartimento di Informatica, Università degli Studi di Verona, Italy
e-mail address: massimo.merro@univr.it

Abstract. We present a timed process calculus for modelling wireless networks in which individual stations broadcast and receive messages; moreover the broadcasts are subject to collisions. Based on a reduction semantics for the calculus we define a contextual equivalence to compare the external behaviour of such wireless networks. Further, we construct an extensional LTS (labelled transition system) which models the activities of stations that can be directly observed by the external environment. Standard bisimulations in this LTS provide a sound proof method for proving systems contextually equivalence. We illustrate the usefulness of the proof methodology by a series of examples. Finally we show that this proof method is also complete, for a large class of systems.

1. Introduction

Wireless networks are becoming increasingly pervasive with applications across many domains, [42, 1]. They are also becoming increasingly complex, with their behaviour depending on ever more sophisticated protocols. There are different levels of abstraction at which these can be defined and implemented, from the very basic level in which the communication primitives consist of sending and receiving electromagnetic signals, to the higher level where the basic primitives allow the initiation of connections between nodes in a wireless system and the exchange of data between them [52].

Assuring the correctness of the behaviour of a wireless network has always been difficult. Several approaches have been proposed to address this issue for networks described at a high level [38, 33, 17, 16, 49, 27, 7, 10]; these typically allow the formal description of protocols at the *network layer* of the *TCP/IP* reference model [52]. However there are few frameworks in the literature which consider networks described at the MAC-Sublayer of the *TCP/IP* reference model

Received by the editors 13th December 2013.

An extended abstract appeared in the proceeding of the *5th International Conference on Coordination Models and Languages* (COORDINATION 2013), volume 7890 of *Lecture Notes in Computer Science*, pp. 16-30, Springer, 2013.

The first and the second authors are supported by SFI project SFI 06 IN.1 1898.

The third author was partially supported by the PRIN 2010-2011 national project "Security Horizons".

LOGICAL METHODS
IN COMPUTER SCIENCE
DOI:10.2168/LMCS-???

[28, 34, 8, 54]. This is the topic of the current paper. We propose a process calculus for describing and verifying wireless networks at the *MAC-Sublayer* of the *TCP/IP* reference model.

This calculus, called the Calculus of Collision-prone Communicating Processes (CCCP), has been largely inspired by TCWS [34]; in particular CCCP inherits its communication features but simplifies considerably the syntax, the reduction semantics, the notion of observation, and as we will see the behavioural theory. In CCCP a wireless system is considered to be a collection of wireless stations which transmit and receive messages. The transmission of messages is *broadcast*, and it is *time-consuming*; the transmission of a message v can require several time slots (or instants). In addition, wireless stations in our calculus are sensitive to *collisions*; if two different stations are transmitting a value over a channel c at the same time slot then a collision occurs; as a result, the content of the messages originally being transmitted is lost.

More specifically, in CCCP a state of a wireless network (or simply network, or system) will be described by a *configuration* of the form $\Gamma \triangleright W$ where W describes the code running at individual wireless stations and Γ represents the communication state of channels. At any given point of time there may be *exposed* communication channels, that is channels containing messages (or values) in transmission; this information will be recorded in Γ.

Such systems evolve by the broadcast of messages between stations, the passage of time, or some other internal activity, such as the occurrence of collisions and their consequences. One of the topics of the paper is to capture formally these complex evolutions, by defining a *reduction semantics*, whose judgements take the form $\Gamma_1 \triangleright W_1 \twoheadrightarrow \Gamma_2 \triangleright W_2$. We show that the reduction semantics we propose satisfies some desirable time properties such as *time determinism*, *maximal progress* and *patience* [39, 22, 56].

However the main aim of the paper is to develop a behavioural theory of wireless networks with time-consuming communications. To this end we need a formal notion of when two such systems are indistinguishable from the point of view of users. Having a reduction semantics it is now straightforward to adapt a standard notion of *contextual equivalence*: $\Gamma_1 \triangleright W_1 \simeq \Gamma_2 \triangleright W_2$. Intuitively this means that either system, $\Gamma_1 \triangleright W_1$ or $\Gamma_2 \triangleright W_2$, can be replaced by the other in a larger system without changing the observable behaviour of the overall system. Formally, we use the approach of [23, 45], often called *reduction barbed congruence*, rather than that of [35][1]. The only parameter in the definition of our contextual equivalence is the choice of primitive observation or *barb*; our choice is natural for wireless systems: the ability to transmit on an idle (or unexposed) channel, that is a channel with no active transmissions.

As explained in papers such as [43, 21], contextual equivalences are determined by so-called *extensional actions*, that is the set of minimal observable interactions which a system can have with its external environment. For CCCP determining these actions is non-trivial. Although values can be transmitted and received on channels, the presence of collisions means that these are not necessarily observable. In fact the important point is not the transmission of a value, but its successful delivery. Also, although the basic notion of observation on systems does not involve the recording of the passage of time, this has to be taken into account extensionally in order to gain a proper extensional account of systems.

The extensional semantics determines an LTS (labelled transition system) over configurations, which in turn gives rise to the standard notion of (weak) bisimulation equivalence between configurations. This gives a powerful co-inductive proof technique: to show that two systems are behaviourally equivalent it is sufficient to exhibit a witness bisimulation which contains them.

[1] See page 106 of [47] for a brief discussion of the difference.

One result of this paper is that weak bisimulation in the extensional LTS is sound with respect to the touchstone contextual equivalence: if two systems are related by some bisimulation in the extensional LTS then they are contextually equivalent. In order to show the effectiveness of our bisimulation proof method we prove a number of non-obvious system equalities. However, the main contribution of the current paper is that completeness holds for a large class of networks, called *well-formed*. If two such networks are contextually equivalent then there is some bisimulation, based on our novel extensional actions, which contains them. In [34], a sound but not complete bisimulation based proof method is developed for (a different form of) reduction barbed congruence. Here, by simplifying the calculus and isolating novel extensional actions we obtain both soundness and completeness.

We end this introduction with an outline of the paper. In Section 2 we present the calculus CCCP. More precisely, Section 2.1 contains the syntax of our language; Section 2.2 introduces the intensional semantics; Section 2.3 provides the reduction semantics; Section 2.4 defines our touchstone contextually-defined behavioural equivalence for comparing wireless networks.

In Section 3 we address the problem of defining the minimal observable activities of systems. These are defined as actions of an extensional semantics in Section 3.1, while in Section 3.2 we consider the bisimulation principle induced by such actions.

In Section 4 we present the main results of the paper. First we prove that our bisimulation proof technique is sound with respect to the contextual equivalence, Section 4.1. In Section 4.2 we prove that, for a large class of configurations, called well-formed, our proof technique is also complete.

The usefulness of our bisimulation proof technique is shown in Section 5, where we consider simple case studies which model common features of wireless networks at the Mac-Layer.

Section 6 concludes the paper with a comparison with the related work.

2. The calculus

As already discusses a wireless system will be represented in our calculus as a *configuration* of the form $\Gamma \triangleright W$, where W describes the code running at individual wireless stations and Γ is a channel environment containing the transmission information for channels. A possible evolution of a system will then be given by a sequence of computation steps:

$$\Gamma_1 \triangleright W_1 \twoheadrightarrow \Gamma_2 \triangleright W_2 \twoheadrightarrow \ldots\ldots \twoheadrightarrow \Gamma_k \triangleright W_k \ldots \twoheadrightarrow \ldots \tag{2.1}$$

where intuitively each step corresponds to either the passage of time, a broadcast from a station, or some unspecified internal computation; the code running at stations evolves as a computation proceeds, but so also does the state of the underlying channel environment. In the following we will use the meta-variable C to range over configurations.

2.1. Syntax.

Formally we assume a set of channels **Ch**, ranged over by $c, d, \cdots$, and a set of values **Val**, which contains a set of data-variables, ranged over by $x, y, \cdots$ and a special value err; this value will be used to denote faulty transmissions. The set of *closed values*, that is those not containing occurrences of variables, are ranged over by $v, w, \cdots$. We also assume that every closed value $v \in$ **Val** has an associated strictly positive integer δ_v, which denotes the number of time slots needed by a wireless station to transmit v. Finally, we assume a language of expressions **Exp** which can be built from values in **Val**; we also assume a function $[\![\cdot]\!]$, for evaluating expressions with no occurrences of data-variables into closed values.

A channel environment is a mapping $\Gamma : \mathbf{Ch} \to \mathbb{N} \times \mathbf{Val}$. In a configuration $\Gamma \triangleright W$ where $\Gamma(c) = (n, v)$ for some channel c, there is a wireless station which is currently transmitting the value

Table 1 CCCP: Syntax

W	::=	P	station code
	$\mid$	$c[x].P$	active receiver
	$\mid$	$W_1 \mid W_2$	parallel composition
	$\mid$	$\nu c{:}(n, v).W$	channel restriction
P, Q	::=	$c\,!\langle e\rangle.P$	broadcast
	$\mid$	$\lfloor c?(x).P \rfloor Q$	receiver with timeout
	$\mid$	$\sigma.P$	delay
	$\mid$	$\tau.P$	internal activity
	$\mid$	$P + Q$	choice
	$\mid$	$[b]P, Q$	matching
	$\mid$	X	process variable
	$\mid$	nil	termination
	$\mid$	fix $X.P$	recursion

Channel Environment: $\Gamma : \mathbf{Ch} \to \mathbb{N} \times \mathbf{Val}$

v for the next n time slots. We will use some suggestive notation for channel environments: $\Gamma \vdash_{\mathrm{t}} c : n$ in place of $\Gamma(c) = (n, w)$ for some w, $\Gamma \vdash_{\mathrm{v}} c : w$ in place of $\Gamma(c) = (n, w)$ for some n. If $\Gamma \vdash_{\mathrm{t}} c : 0$ we say that channel c is idle in Γ, and we denote it with $\Gamma \vdash c : \mathbf{idle}$. Otherwise we say that c is exposed in Γ, denoted by $\Gamma \vdash c : \mathbf{exp}$. The channel environment Γ such that $\Gamma \vdash c : \mathbf{idle}$ for every channel c is said to be *stable*. Often we will compare channel environments according to the amount of time instants for which channels will be exposed; we say that $\Gamma \leq \Gamma'$ if, for any channel c, $\Gamma \vdash_{\mathrm{t}} c : n$ implies $\Gamma' \vdash_{\mathrm{t}} c : m$, for some m such that $n \leq m$.

The syntax for system terms W is given in Table 1, where P ranges over code for programming individual stations, which is also illustrated in Table 1. A system term W is a collection of individual threads running in parallel, with possibly some channels restricted.

Each thread may be either an inactive piece of code P or an active code of the form $c[x].P$. This latter term represents a wireless station which is receiving a value from the channel c; when the value is eventually received the variable x will be replaced with the received value in the code P.

The syntax for station code is based on standard process calculus constructs. The main constructs are time-dependent reception from a channel $\lfloor c?(x).P \rfloor Q$, explicit time delay $\sigma.P$, and broadcast along a channel $c\,!\langle e\rangle.P$; here the value being broadcast is the one obtained by evaluating e via the function $[\![\cdot]\!]$, provided that e does not contain any occurrence of data-variables. Of the remaining standard constructs the most notable is matching, $[b]P, Q$ which branches to P or Q, depending on the value of the Boolean expression b. Such boolean expressions can be either equality tests of the form $e_1 = e_2$, or terms of the form $\mathrm{exp}(c)$, which will be used to check whether channel c is exposed, that is it is being used for transmission.

In the construct fix $X.P$ occurrences of the recursion variable X in P are bound; similarly in the terms $\lfloor c?(x).P \rfloor Q$ and $c[x].P$ the data-variable x is bound in P. This gives rise to the standard notions

of free and bound variables, α-conversion and capture-avoiding substitution; In a configuration of the form $\Gamma \triangleright W$, we assume that W is closed, meaning that all its occurrences of both data-variables and process variables are bound. In general, we always assume that a system term W is closed, unless otherwise stated. Sometimes we will need to consider system terms with free occurrences of process variables, we will explicitly say that they are open system terms. System terms, both open and closed, are identified up to α-conversion. We assume that all occurrences of recursion variables are *guarded*; they must occur within either a broadcast, input residual, timeout branch, time delay prefix, or within an execution branch of a matching construct. This ensures that recursive calls cannot be used to build up infinite loops within a time slot

Example 2.1. Consider the configuration

$$C_1 = \Gamma \triangleright S_1 \mid S_2 \mid R_1$$

where

$$\begin{aligned} S_1 &= c\,!\langle v_0\rangle.\mathsf{nil} \\ S_2 &= \sigma.c\,!\langle v_1\rangle.\mathsf{nil} \\ R_1 &= \lfloor c?(x).P\rfloor\mathsf{nil} \end{aligned}$$

and Γ is the stable channel environment. Further, we assume that $\delta_{v_0} = 2$ and $\delta_{v_1} = 1$. This configuration contains two sender stations, running the code S_1 and S_2, respectively, and a receiving station, running the code R_1. In the first time slot, the station running the code S_1 broadcasts the value v_0 along channel c. The station running the code R_1 starts receiving such a value and it will be busy in receiving it for the next two time slots. In the first time slot the station running the code S_2 is idle. It is only in the second time slot that this station will broadcast a value along channel c. At this point the receiving station will be exposed to two transmissions; the transmission of value v_0, which is still in progress, and the transmission of value v_1. As a result, a collision happens, and the value received by the receiver will be at the end error value err.

The formal behaviour of the configuration C_1 will be explained in Example 2.17. □

We use a number of notational conventions. $\prod_{i\in I} W_i$ means the parallel composition of all stations W_i, for $i \in I$. We identify $\prod_{i\in I} W_i$ with nil if $I = \emptyset$. We will omit trailing occurrences of nil, render $\nu c\,{:}(n, v).W$ as $\nu c.W$ when the values (n, v) are not relevant to the discussion, and use $\nu\tilde{c}.W$ as an abbreviation for a sequence $\tilde{c}$ of such restrictions. We write $\lfloor c?(x).P\rfloor$ for $\lfloor c?(x).P\rfloor\mathsf{nil}$. Finally, we abbreviate the recursive process $\mathsf{fix}\, X.\lfloor c?(x).P\rfloor X$ with $c?(x).P$; as we will see this is a persistent listener at channel c waiting for an incoming message.

2.2. **Intensional semantics.** Our first goal is to formally define computation steps among configurations of the form $\Gamma_1 \triangleright W_1 \twoheadrightarrow \Gamma_2 \triangleright W_2$. In order to do that, we first define the evolution of system terms with respect to a channel environment Γ via a set of SOS rules whose judgements take the form $\Gamma \triangleright W_1 \xrightarrow{\lambda} W_2$, were λ is an intensional action taking one of the following forms:

(1) $c!v$, denoting a station starting broadcasting value v along channel c
(2) σ, denoting the passage of one time slot, or time instant
(3) τ, denoting an internal action
(4) $c?v$, denoting a station in the external environment starting broadcasting value v on channel c.

These actions λ will have an effect also on the channel environment, which we describe by means of a functional $\mathrm{upd}_\lambda(\cdot) : \mathbf{Env} \to \mathbf{Env}$, where $\mathbf{Env}$ is the set of channel environments.

Table 2 Intensional semantics: transmission

$$\text{(Snd)}\ \frac{[\![e]\!] = v}{\Gamma \triangleright c\,!\langle e\rangle.P \xrightarrow{c!v} \sigma^{\delta_v}.P} \qquad \text{(Rcv)}\ \frac{\Gamma \vdash c : \mathbf{idle}}{\Gamma \triangleright \lfloor c?(x).P \rfloor Q \xrightarrow{c?v} c[x].P}$$

$$\text{(RcvIgn)}\ \frac{\neg \mathsf{rcv}(\Gamma \triangleright W, c)}{\Gamma \triangleright W \xrightarrow{c?v} W} \qquad \text{(Sync)}\ \frac{\Gamma \triangleright W_1 \xrightarrow{c!v} W_1' \quad \Gamma \triangleright W_2 \xrightarrow{c?v} W_2'}{\Gamma \triangleright W_1 \mid W_2 \xrightarrow{c!v} W_1' \mid W_2'}$$

$$\text{(RcvPar)}\ \frac{\Gamma \triangleright W_1 \xrightarrow{c?v} W_1' \quad \Gamma \triangleright W_2 \xrightarrow{c?v} W_2'}{\Gamma \triangleright W_1 \mid W_2 \xrightarrow{c?v} W_1' \mid W_2'}$$

Definition 2.2. [Channel Environment update] Let $\Gamma \in \mathbf{Env}$ be an arbitrary channel environment and $c \in \mathbf{Ch}$ an arbitrary channel. Let t_c and v_c be the exposure time and the value transmitted along channel c in Γ, respectively, that is $\Gamma \vdash_{\mathrm{t}} c : t_c$ and $\Gamma \vdash_{\mathrm{v}} c : v_c$. For any intensional action λ, we let $\mathrm{upd}_\lambda(\Gamma)$ be the unique channel environment determined by the following definitions:[2]

(1) $\mathrm{upd}_\sigma(\Gamma) \vdash_{\mathrm{t}} c : t_c - 1$ and $\mathrm{upd}_\sigma(\Gamma) \vdash_{\mathrm{v}} c : v_c$;
(2) for any value $v \in \mathbf{Val}$, let $\mathrm{upd}_{c!v}(\Gamma)$ be the channel environment such that

$$\mathrm{upd}_{c!v}(\Gamma) \vdash_{\mathrm{t}} c : \begin{cases} \delta_v & \text{if } \Gamma \vdash c : \mathbf{idle} \\ \max(\delta_v, t_c) & \text{if } \Gamma \vdash c : \mathbf{exp} \end{cases} \qquad \mathrm{upd}_{c!v}(\Gamma) \vdash_{\mathrm{v}} c : \begin{cases} v & \text{if } \Gamma \vdash c : \mathbf{idle} \\ \mathsf{err} & \text{if } \Gamma \vdash c : \mathbf{exp} \end{cases}$$

and for any channel d, $d \neq c$, let $\mathrm{upd}_{c!v}(\Gamma) \vdash_{\mathrm{t}} d : t_d$ and $\mathrm{upd}_{c!v}(\Gamma) \vdash_{\mathrm{v}} d : v_d$;
(3) for any value v, $\mathrm{upd}_{c?v}(\Gamma) = \mathrm{upd}_{c!v}(\Gamma)$;
(4) $\mathrm{upd}_\tau(\Gamma) = \Gamma$.

□

Let us describe the intuitive meaning of this definition. When time passes, the time of exposure of each channel decreases by one time unit. The predicates $\mathrm{upd}_{c!v}(\Gamma)$ and $\mathrm{upd}_{c?v}(\Gamma)$ model how collisions are handled in our calculus. When a station begins broadcasting a value v over an idle channel c this channel becomes exposed for the amount of time required to transmit v, that is δ_v. If the channel is not idle a collision happens. As a consequence, the value that will be received by a receiving station, when all transmissions over channel c terminate, is the error value err, and the exposure time is adjusted accordingly. Finally the definition of $\mathrm{upd}_\tau(\Gamma)$ reflects the intuition that internal activities do not affect the exposure state of channels.

Let us turn our attention to the intensional semantics of system terms. For the sake of clarity, the inference rules for the evolution of system terms, $\Gamma \triangleright W_1 \xrightarrow{\lambda} W_2$, are split in four tables, each one focusing on a particular form of activity.

Table 2 contains the rules governing transmission. Rule (Snd) models a non-blocking broadcast of a message along channel c. The value v sent by process $c\,!\langle e\rangle.P$ is the one obtained by evaluating an expression e; note that here we are assuming that e is closed, hence we can evaluate it to a closed value via the function $[\![\cdot]\!]$. A transmission can fire at any time, independently on the state of the network; the notation σ^{δ_v} represents the time delay operator σ iterated δ_v times. So when the process $c\,!\langle v\rangle.P$ broadcasts, it has to wait δ_v time units (the time required to transmit v) before the residual P is activated. On the other hand, reception of a message by a time-guarded listener

[2]For convenience we assume $0 - 1$ to be 0.

$\lfloor c?(x).P \rfloor Q$ depends on the state of the channel environment. If the channel c is free then rule (Rcv) indicates that reception can start and the listener evolves into the active receiver $c[x].P$.

Rule (RcvIgn) states that if a system term W is not waiting for a message along a channel c, or if c is already exposed, then any broadcast along c is ignored by the configuration $\Gamma \triangleright W$. Here $\mathsf{rcv}(\Gamma \triangleright W, c)$ is a predicate which evaluates to true in the case that in $\Gamma \triangleright W$ channel c is not exposed, and W contains among its parallel components at least one non-guarded receiver of the form $\lfloor c?(x).P \rfloor Q$ which is actively awaiting a message. Formally, we first define a predicate $\mathsf{rcv}(W, c)$ for open terms, which is then lifted to configurations. For open terms we have $\mathsf{rcv}(\Gamma \triangleright W, c)$ is defined inductively as

$$\begin{aligned}
\mathsf{rcv}(P, c) = \text{false} \quad & \text{provided} \quad P = c\,!\langle e\rangle.Q, P = \tau.Q \text{ or } P = X \\
\mathsf{rcv}(\lfloor d?(x).P \rfloor Q, c) = \text{true} \quad & \text{if and only if} \quad d = c \\
\mathsf{rcv}(P + Q, c) = \text{true} \quad & \text{if and only if} \quad \mathsf{rcv}(P, c) = \text{true and } \mathsf{rcv}(Q, c) = \text{true} \\
\mathsf{rcv}(\mathsf{fix}\, X.P, c) = \text{true} \quad & \text{if and only if} \quad \mathsf{rcv}(P, c) = \text{true}
\end{aligned}$$

$$\begin{aligned}
\mathsf{rcv}(c[x].P, d) = \text{false} \quad & \text{always} \\
\mathsf{rcv}(W_1 \mid W_2, c) = \text{true} \quad & \text{if and only if} \quad \mathsf{rcv}(W_1, c) = \text{true and } \mathsf{rcv}(W_2, c) = \text{true} \\
\mathsf{rcv}(\nu d.W, c) = \text{true} \quad & \text{if and only if} \quad \mathsf{rcv}(W, c) = \text{true, where we assume} d \neq c
\end{aligned}$$

Then, for any configuration $\Gamma \triangleright W$, we let $\mathsf{rcv}(\Gamma \triangleright W, c) = \text{true}$ if and only if $\Gamma \vdash c : \mathbf{idle}$ and $\mathsf{rcv}(W, c) = \text{true}$.

The remaining two rules in Table 2 (Sync) and (RcvPar) serve to synchronise parallel stations on the same transmission [20, 39, 40].

Example 2.3. [Transmission] Let $C_0 = \Gamma_0 \triangleright W_0$, where $W_0 = c!\langle v_0\rangle \mid \lfloor d?(x).\mathsf{nil} \rfloor(\lfloor c?(x).Q \rfloor) \mid \lfloor c?(x).P \rfloor$, with $\delta_{v_0} = 2$, and Γ_0 a stable environment.

Using rule (Snd) we can infer $\Gamma_0 \triangleright c!\langle v_0\rangle \xrightarrow{c!v_0} \sigma^2$; this station starts transmitting the value v_0 along channel c. Rule (RcvIgn) can be used to derive the transition $\Gamma_0 \triangleright \lfloor d?(x).\mathsf{nil} \rfloor(\lfloor c?(x).Q \rfloor) \xrightarrow{c?v_0} \lfloor d?(x).\mathsf{nil} \rfloor(\lfloor c?(x).Q \rfloor)$, in which the broadcast of value v_0 along channel c is ignored. On the other hand, Rule (RcvIgn) cannot be applied to the configuration $\Gamma_0 \triangleright \lfloor c?(x).P \rfloor$, since this station is waiting to receive a value on channel c; however we can derive the transition $\Gamma_0 \triangleright \lfloor c?(x).P \rfloor \xrightarrow{c?v_0} c[x].P$ using Rule (Rcv).

We can put together the three transitions above using the rule (Sync), leading to the transition $C_0 \xrightarrow{c!v} W_1$, where $W_1 = \sigma^2 \mid \lfloor d?(x).\mathsf{nil} \rfloor(\lfloor c?(x).Q \rfloor) \mid c[x].P$. □

Example 2.4. [Ignored Receptions] Consider the configuration $C = \Gamma \triangleright c!\langle v\rangle \mid \lfloor c?(x).P \rfloor Q$, where $\delta_v = 1$ and Γ is such that $\Gamma \vdash c : \mathbf{exp}$, say $\Gamma \vdash_{\mathrm{t}} c : 1$. Using the rules introduced so far we can derive

$$C \xrightarrow{c!v} \Gamma \triangleright \sigma \mid \lfloor c?(x).P \rfloor Q \tag{2.2}$$

describing the unblocked sending of the value v along the channel c. This can be inferred using Rule (Sync) from $\Gamma \triangleright c!\langle v\rangle \xrightarrow{c!v} \sigma$, which can be inferred using Rule (Snd), and the judgement $\Gamma \triangleright \lfloor c?(x).P \rfloor Q \xrightarrow{c?v} \lfloor c?(x).P \rfloor Q$. This latter can be inferred using Rule (RcvIgn), because $\Gamma \vdash c : \mathbf{exp}$ means that $\mathsf{rcv}(\Gamma \triangleright \lfloor c?(x).P \rfloor Q, c) = \text{false}$.

In the transition (2.2) above the receiver $\lfloor c?(x).P \rfloor Q$ ignores the transmission of v along c. One might have expected it to accept this value. However the channel is already exposed, $\Gamma \vdash c : \mathbf{exp}$,

Table 3 Intensional semantics: timed transitions

$$\text{(TimeNil)}\ \frac{}{\Gamma \triangleright \mathsf{nil} \xrightarrow{\sigma} \mathsf{nil}} \qquad \text{(Sleep)}\ \frac{}{\Gamma \triangleright \sigma.P \xrightarrow{\sigma} P}$$

$$\text{(ActRcv)}\ \frac{\Gamma \vdash_{\mathrm{t}} c : n,\ n > 1}{\Gamma \triangleright c[x].P \xrightarrow{\sigma} c[x].P} \qquad \text{(EndRcv)}\ \frac{\Gamma \vdash_{\mathrm{t}} c : 1,\ \Gamma \vdash_{\mathrm{v}} c = w}{\Gamma \triangleright c[x].P \xrightarrow{\sigma} \{^{w}/_{x}\}P}$$

$$\text{(Timeout)}\ \frac{\Gamma \vdash c : \mathbf{idle}}{\Gamma \triangleright \lfloor c?(x).P \rfloor Q \xrightarrow{\sigma} Q}$$

Table 4 Intensional semantics: - internal activity

$$\text{(RcvLate)}\ \frac{\Gamma \vdash c : \mathbf{exp}}{\Gamma \triangleright \lfloor c?(x).P \rfloor Q \xrightarrow{\tau} c[x].\{^{\mathsf{err}}/_{x}\}P} \qquad \text{(Tau)}\ \frac{}{\Gamma \triangleright \tau.P \xrightarrow{\tau} P}$$

$$\text{(Then)}\ \frac{[\![b]\!]_{\Gamma} = \text{true}}{\Gamma \triangleright [b]P, Q \xrightarrow{\tau} \sigma.P} \qquad \text{(Else)}\ \frac{[\![b]\!]_{\Gamma} = \text{false}}{\Gamma \triangleright [b]P, Q \xrightarrow{\tau} \sigma.Q}$$

and thus the receptor can not properly synchronise properly with the sender. We will see later, in Example 2.7, that a transmission errors actually occurs. □

The transitions for modelling the passage of time, $\Gamma \triangleright W \xrightarrow{\sigma} W'$, are given in Table 3. Rules (TimeNil) and (Sleep) are straightforward. In rules (ActRcv) and (EndRcv) we see that the active receiver $c[x].P$ continues to wait for the transmitted value to make its way through the network; when the allocated transmission time elapses the value is then delivered and the receiver evolves to $\{^{w}/_{x}\}P$. Finally, Rule (Timeout) implements the idea that $\lfloor c?(x).P \rfloor Q$ is a time-guarded receptor; when time passes it evolves into the alternative Q. However this only happens if the channel c is not exposed. What happens if it is exposed is explained in Table 4.

Example 2.5. [Passage of Time] Let $C_1 = \Gamma_1 \triangleright W_1$, where $\Gamma_1(c) = (2, v_0)$, $\Gamma_1 \vdash d$: **idle** and $W_1 = \sigma^2 \mid \lfloor d?(x).\mathsf{nil} \rfloor \lfloor c?(x).Q \rfloor \mid c[x].P$ is the system term derived in Example 2.3. We show how a σ-action can be derived for this configuration. First note that $\Gamma_1 \triangleright \sigma^2 \xrightarrow{\sigma} \sigma$; this transition can be derived using Rule (Sleep). Since d is idle in Γ_1, we can apply Rule (TimeOut) to infer the transition $\Gamma_1 \triangleright \lfloor d?(x).\mathsf{nil} \rfloor (\lfloor c?(x).Q \rfloor) \xrightarrow{\sigma} \lfloor c?(x).Q \rfloor$; time passed before a value could be broadcast along channel d, causing a timeout in the station waiting to receive a value along d. Finally, since $\Gamma_1 \vdash_{\mathrm{v}} c : 2$, we can use Rule (ActRcv) to derive $\Gamma_1 \triangleright c[x].P \xrightarrow{\sigma} c[x].P$.

At this point we can use twice Rule (TimePar) (which is given in Table 5) to infer a σ-action performed by C_1. This leads to the transition $C_1 \xrightarrow{\sigma} W_2$, where $W_2 = \sigma \mid \lfloor c?(x).Q \rfloor \mid c[x].P$. □

Table 4 is devoted to internal transitions $\Gamma \triangleright W \xrightarrow{\tau} W'$. Let us first explain rule (RcvLate). Intuitively the process $\lfloor c?(x).P \rfloor Q$ is ready to start receiving a value on channel c. However if c is exposed this means that a transmission is already taking place. Since the process has therefore missed the start of the transmission it will receive an error value. Thus the rule (RcvLate) reflects the fact that in wireless systems a collision takes place if there is a misalignment between the transmission

and reception of a message. The remaining rules are straightforward. Note that in the matching construct we use a channel environment dependent evaluation function for Boolean expressions $[\![b]\!]_\Gamma$ (note that this has not to be confused with the function $[\![\cdot]\!]$, used to evaluate closed expressions), because of the presence of the exposure predicate $\exp(c)$ in the Boolean language. Formally we have that $[\![e_1 = e_2]\!]_\Gamma$ = true evaluates to true if and only if $[\![e_1]\!] = [\![e_2]\!]$, and $[\![\exp(c)]\!]_\Gamma$ = true if and only if $\Gamma \vdash c : \mathbf{exp}$. We remark that checking for the exposure of a channel amounts to listening on the channel for a value. But in wireless systems it is not possible to both listen and transmit within the same time unit, as communication is half-duplex, [42]. As a consequence in our intensional semantics, in the rules (Then) and (Else), the execution of both branches is delayed of one time unit.

Example 2.6. Let Γ_2 be a channel environment such that $\Gamma_2(c) = (1, v)$, and consider the configuration $C_2 = \Gamma_2 \triangleright W_2$, where $W_2 = \sigma \mid \lfloor c?(x).Q \rfloor \mid c[x].P$ has been defined in Example 2.5.

Note that this configuration contains both a receiver process and an active receiver along the exposed channel c. We can think of the receiver $\lfloor c?(x).Q \rfloor$ as a process which missed the synchronisation with a broadcast which has been previously performed along channel c; as a consequence this process is doomed to receive an error value.

This situation is modelled by Rule (RcvLate), which allows us to infer the transition $\Gamma_2 \triangleright \lfloor c?(x).Q \rfloor \xrightarrow{\tau} c[x].\{\mathsf{err}/x\}Q$. As we will see, Rule (TauPar) which we introduce in Table 5, ensures that τ-actions are contextual. This means that the transition derived above allows us to infer the transition $C_2 \xrightarrow{\tau} W_3$, where $W_3 = \sigma \mid c[x].\{\mathsf{err}/x\}Q \mid c[x].P$. □

Example 2.7. [On rules (RcvIgn) and (RcvLate)] Consider again the configuration C of Example 2.4. Recall that $C = \Gamma \triangleright c!\langle v \rangle \mid \lfloor c?(x).P \rfloor Q$, where $\Gamma \vdash_{\mathrm{v}} c : 1$ and $\delta_v = 1$. In Example 2.4 we have shown that $C \xrightarrow{c!v} \sigma \mid \lfloor c?(x).P \rfloor Q$, where the proof of the transition contains an application of Rule (RcvIgn). This transition represents the unblocked transmission of the value v along the channel c, which also changes the channel environment from Γ to $\mathrm{upd}_{c!v}(\Gamma)$. Now consider the resulting configuration $C' = \mathrm{upd}_{c!v}(\Gamma) \triangleright \sigma \mid \lfloor c?(x).P \rfloor Q$. As $\mathrm{upd}_{c!v}(\Gamma) \vdash c : \mathbf{exp}$ we can use Rule (RcvLate)[3], to infer the transition $C' \xrightarrow{\tau} \sigma \mid c[x].\{\mathsf{err}/x\}P$, modelling the expected error in transmission along channel c due to a collision.

Note also that we could have applied Rule (RcvLate) directly to the initial configuration $C = \Gamma \triangleright c!\langle v \rangle \mid \lfloor c?(x).P \rfloor Q$, leading to the transition $C \xrightarrow{\tau} c!\langle v \rangle \mid c[x].\{\mathsf{err}/x\}P$, again reflecting an error in transmission along the channel c due to the fact that it is already exposed. In fact we have the transition $\Gamma \triangleright W \mid \lfloor c?(x).P \rfloor Q \xrightarrow{\tau} W \mid c[x].\{\mathsf{err}/x\}P$, regardless of the form of W. This emphasises the fact that the inability of the receiver to receive correctly the value being transmitted is because the channel is already exposed and not because another station is willing to broadcast along it. □

Remark 2.8. The previous example together with Example 2.4 shows that there is a delicate interplay between the rules (RcvIgn) and (RcvLate), particularly when modelling the effect of an external broadcast on receivers in the presence of exposed channels. The overall goal of our intensional semantics is to ensure that it has certain natural properties, such as *input-enabledness*. This ensures that for any configuration $\Gamma \triangleright W$ and any $c?v$ there exists some transition $\Gamma \triangleright W \xrightarrow{c?v} W'$. Here W' records the effect of an external broadcast of v along c has on the configuration; if the broadcast is actually ignored by all stations in the configuration then W' will coincide with W. *Input-enabledness* also helps us in ensuring that broadcasts are independent of their environment. For example, we require the configuration $(\Gamma \triangleright c!\langle v \rangle \mid W)$ to be able to perform the broadcast of value v along channel c, regardless of the structure of W, even if c is exposed in Γ. Such a transition can only be inferred

[3]An application of Rule (TauPar) from Table 5 is also required.

Table 5 Intensional semantics: - structural rules

$$\text{(TimePar)}\ \frac{\Gamma \triangleright W_1 \xrightarrow{\sigma} W_1' \quad \Gamma \triangleright W_2 \xrightarrow{\sigma} W_2'}{\Gamma \triangleright W_1 \mid W_2 \xrightarrow{\sigma} W_1' \mid W_2'} \qquad \text{(TauPar)}\ \frac{\Gamma \triangleright W_1 \xrightarrow{\tau} W_1'}{\Gamma \triangleright W_1 \mid W_2 \xrightarrow{\tau} W_1' \mid W_2}$$

$$\text{(Rec)}\ \frac{\{\mathsf{fix}\, X.P/X\}P \xrightarrow{\lambda} W}{\Gamma \triangleright \mathsf{fix}\, X.P \xrightarrow{\lambda} W} \qquad \text{(Sum)}\ \frac{\Gamma \triangleright P \xrightarrow{\lambda} W \quad \lambda \in \{\tau, c!v\}}{\Gamma \triangleright P + Q \xrightarrow{\lambda} W}$$

$$\text{(SumTime)}\ \frac{\Gamma \triangleright P \xrightarrow{\sigma} P' \quad \Gamma \triangleright Q \xrightarrow{\sigma} Q'}{\Gamma \triangleright P + Q \xrightarrow{\sigma} \Gamma' \triangleright P' + Q'} \qquad \text{(SumRcv)}\ \frac{\Gamma \triangleright P \xrightarrow{c?v} W \quad \mathsf{rcv}(\Gamma \triangleright P, c)}{\Gamma \triangleright P + Q \xrightarrow{c?v} W}$$

$$\text{(ResI)}\ \frac{\Gamma[c \mapsto (n,v)] \triangleright W \xrightarrow{c!v} W'}{\Gamma \triangleright \nu c{:}(n,v).W \xrightarrow{\tau} \nu c{:}\mathrm{upd}_{c!v}(\Gamma)(c).W'} \qquad \text{(ResV)}\ \frac{\Gamma[c \mapsto (n,v)] \triangleright W \xrightarrow{\lambda} W', \ c \notin \lambda}{\Gamma \triangleright \nu c{:}(n,v).W \xrightarrow{\lambda} \nu c{:}(n,v).W}$$

from Rule (Sync) if we we match the output action along channel c performed by the configuration $\Gamma \triangleright c!\langle v\rangle$ with an input action performed by $\Gamma \triangleright W$. *Input-enabledness* will ensure that the latter input action is always possible.

In Section 2.2 we will show that our intensional semantics in fact satisfies a number of natural properties, including *input-enabledness*; see Lemma 2.9. This would obviously be not true if, by omitting Rule (Rcvlgn), we were to forbid inputs over exposed channels. □

The final set of rules, in Table 5, are structural. Rule (TimePar) models how σ-actions are derived for collections of threads. Rules (TauPar), (Rec) and (Sum) are standard. Rule (SumTime) is necessary to ensure *time determinism* (see Proposition 2.10). Rule (SumRcv) guaranteed that only effective receptions can decide in a choice process. Finally Rules (ResI) and (ResV) show how restricted channels are handled. Intuitively moves from the configuration $\Gamma \triangleright \nu c:(n,v).W$ are inherited from the configuration $\Gamma[c \mapsto (n,v)] \triangleright W$; here the channel environment $\Gamma[c \mapsto (n,v)]$ is the same as Γ except that c has associated with it (temporarily) the information (n,v). However if this move mentions the restricted channel c then the inherited move is rendered as an internal action τ, (ResI). Moreover the information associated with the restricted channel in the residual is updated, using the function $\mathrm{upd}_{c!v}(\cdot)$ previously defined. Rules (TauPar), (Sum) and (SumRcv) have their symmetric counterparts.

In the remainder of this section we illustrate some of the main properties enjoyed by the intensional semantics illustrated in Section 2.2. The contents of this part are purely technical and needed only for the proofs of the results illustrated later in the paper: they may be safely skipped by the reader not interested in details.

In broadcast process calculi transmission of a value is usually modelled as a non-blocking action [40, 34, 10], meaning that all configurations should always be able to receive an arbitrary value along an arbitrary channel. This is a derived property of our calculus:

Lemma 2.9. [Input enabledness] Let $\Gamma \triangleright W$ be a configuration. Then for any channel c and value v we have that $\Gamma \triangleright W \xrightarrow{c?v} W'$ for some W'; further

(1) $\neg\mathsf{rcv}(\Gamma \triangleright W, c)$ implies $W' = W$

(2) $\mathsf{rcv}(\Gamma \triangleright W, c)$ implies $W' \neq W$, and for every value w, $\Gamma \triangleright W \xrightarrow{c?w} W'$.

Proof. See the Appendix, Page 47. □

Our model of time also conforms to a well-established approach in the literature; see for example [39, 56]:

Proposition 2.10. [Time Determinism] Suppose $C \xrightarrow{\sigma} W_1$ and $C \xrightarrow{\sigma} W_2$. Then $W_1 = W_2$.

Proof. By induction on the proof of the transition $C \xrightarrow{\sigma} W_1$. See the Appendix, Page 49 for details. □

Proposition 2.11. [Maximal Progress] Suppose $C \xrightarrow{\sigma} W_1$. If $\lambda \in \{\tau, c!v\}$, for some c and v, then there is no W_2 such that $C \xrightarrow{\lambda} W_2$.

Proof. By induction on the proof of the derivation $C \xrightarrow{\sigma} W_1$. See the Appendix, Page 49 for details. □

Another important property concerns the exposure state of channel environments. This property states that non-timed transitions are identified up-to channel environments which share the same set of idle channels.

Proposition 2.12. [Exposure Consistency] Let Γ_1, Γ_2 be two channel environments such that $\Gamma_1 \vdash c : \mathbf{exp}$ if and only if $\Gamma_2 \vdash c : \mathbf{exp}$ for every channel c. Then for any system term W and action $\lambda \neq \sigma$, $\Gamma_1 \triangleright W \xrightarrow{\lambda} W'$ implies $\Gamma_2 \triangleright W \xrightarrow{\lambda} W'$.

Proof. By Induction on the proof of the derivation $\Gamma_1 \triangleright W \xrightarrow{\lambda} W'$. See the Appendix, Page 50 for details. □

We end our discussion on the intensional semantics with a technical result on the interaction between stations in systems; this will be useful in later developments.

Proposition 2.13. [Parallel components] Let $\Gamma \triangleright W_1 \mid W_2$ be a configuration.

(1) $\Gamma \triangleright W_1 \mid W_2 \xrightarrow{\tau} W$ if and only if
   - either there is W_1' such that $\Gamma \triangleright W_1 \xrightarrow{\tau} W_1'$ with $W = W_1' \mid W_2$
   - or there is W_2' such that $\Gamma \triangleright W_2 \xrightarrow{\tau} W_2'$ with $W = W_1 \mid W_2'$.

(2) $\Gamma \triangleright W_1 \mid W_2 \xrightarrow{c?v} W$ if and only if there are W_1' and W_2' such that $\Gamma \triangleright W_1 \xrightarrow{c?v} W_1'$, $\Gamma \triangleright W_2 \xrightarrow{c?v} W_2'$ and $W = W_1' \mid W_2'$.

(3) $\Gamma \triangleright W_1 \mid W_2 \xrightarrow{c!v} W$ if and only if there are W_1' and W_2' such that
   - $\Gamma \triangleright W_1 \xrightarrow{c!v} W_1'$, $\Gamma \triangleright W_2 \xrightarrow{c?v} W_2'$ and $W = W_1' \mid W_2'$
   - or $\Gamma \triangleright W_1 \xrightarrow{c?v} W_1'$, $\Gamma \triangleright W_2 \xrightarrow{c!v} W_2'$ and $W = W_1' \mid W_2'$.

(4) $\Gamma \triangleright W_1 \mid W_2 \xrightarrow{\sigma} W$ if and only if there are W_1' and W_2' such that $\Gamma \triangleright W_1 \xrightarrow{\sigma} W_1'$, $\Gamma \triangleright W_2 \xrightarrow{\sigma} W_2'$ and $W = W_1' \mid W_2'$. □

Proof. Details for (3) are given in the Appendix; see Page 50. The other three statements can be proved similarly. □

2.3. **Reduction semantics.** We are now in a position to formally define the individual computation steps for wireless systems, alluded to informally in (2.1) above.

Definition 2.14. [Reduction] We write $\Gamma \triangleright W \twoheadrightarrow \Gamma' \triangleright W'$ if

(i) (Transmission) $\Gamma \triangleright W \xrightarrow{c!v} W'$ for some channel c and value v, where $\Gamma' = \mathrm{upd}_{c!v}(\Gamma)$

(ii) (Time) $\Gamma \triangleright W \xrightarrow{\sigma} W'$ and $\Gamma' = \mathrm{upd}_{\sigma}(\Gamma)$

(iii) (Internal) $\Gamma \triangleright W \xrightarrow{\tau} W'$ and $\Gamma' = \mathrm{upd}_{\tau}(\Gamma)$.

The intuition here should be obvious; computation proceeds either by the transmission of values between stations, the passage of time, or internal activity; further, the exposure state of channels is updated according to the performed transition. □

Sometimes it will be useful to distinguish between instantaneous reductions and timed reductions; instantaneous reductions, $\Gamma_1 \triangleright W_1 \twoheadrightarrow_i \Gamma_2 \triangleright W_2$, are those derived via clauses (i) or (iii) above; timed reductions are denoted with the symbol $\twoheadrightarrow_\sigma$ and coincide with reductions derived using clause (ii). We use the notation $\Gamma \triangleright W \twoheadrightarrow_i (\Gamma \triangleright W \twoheadrightarrow_\sigma)$ if there exists $\Gamma' \triangleright W'$ such that $\Gamma \triangleright W \twoheadrightarrow_i \Gamma' \triangleright W'$ ($\Gamma \triangleright W \twoheadrightarrow_\sigma \Gamma' \triangleright W'$), and $\Gamma \triangleright W \not\twoheadrightarrow_i (\Gamma \triangleright W \not\twoheadrightarrow_\sigma)$ to stress that there is no configuration $\Gamma' \triangleright W'$ such that $\Gamma \triangleright W \twoheadrightarrow_i \Gamma' \triangleright W'$ ($\Gamma \triangleright W \twoheadrightarrow_i \Gamma' \triangleright W'$).

Example 2.15. We show how the transitions we have inferred in the Examples 2.3, 2.5 and 2.6 can be combined together to derive a computation fragment for the configuration C_0 considered in Example 2.3.

Let $C_i = \Gamma_i \triangleright W_i$, $i \in 0, .., 2$, be as defined in the examples mentioned above. Note that $\Gamma_1 = \mathrm{upd}_{c!v_0}(\Gamma_0)$ and $\Gamma_2 = \mathrm{upd}_{\sigma}(\Gamma_1)$. We have already shown that $C_0 \xrightarrow{c!v_0} W_1$; this transition, together with the equality $\Gamma_1 = \mathrm{upd}_{c!v_0}(\Gamma_0)$, can be used to infer the reduction $C_0 \twoheadrightarrow_i C_1$. A similar argument shows that $C_1 \twoheadrightarrow_\sigma C_2$. Finally, if we let C_3 denote $\Gamma_2 \triangleright W_3$ we also have $C_2 \twoheadrightarrow_i C_3$ since $C_2 \xrightarrow{\tau} W_3$ and $\Gamma_2 = \mathrm{upd}_{\tau}(\Gamma_2)$. □

Example 2.16. [Time-consuming transmission] Consider a wireless system with two stations, that is a configuration C_1 of the form $\Gamma_1 \triangleright P_1 \mid Q_1$. Let us suppose

$$P_1 \text{ is } c\,!\langle w\rangle.R, \qquad Q_1 \text{ is } \lfloor c?(x).S \rfloor T_1$$

where Γ_1 is a stable channel environment and $\delta_w = 2$. Then

$$C_1 \twoheadrightarrow C_2 \tag{2.3}$$

where C_2 has the form $\Gamma_2 \triangleright P_2 \mid Q_2$ and

$$P_2 \text{ is } \sigma^2.R \qquad Q_2 \text{ is } c[x].S \qquad \Gamma_2 \vdash_{\mathrm{t}} c : 2 \qquad \Gamma_2 \vdash_{\mathrm{v}} c : w$$

The move from P_1 to P_2 is via an application of the rule (Snd), from Q_1 to Q_2 relies on (Rcv) and they are combined together using (Sync) to obtain $\Gamma_1 \triangleright P_1 \mid Q_1 \xrightarrow{c!w} P_2 \mid Q_2$. The final step (2.3) results from (Transmission) in Definition 2.14.

The next step $C_2 \twoheadrightarrow C_3 = \Gamma_3 \triangleright \sigma.R \mid Q_2$ is via (Time) in Definition 2.14; here the only change to the channel environment is that $\Gamma_3 \vdash_{\mathrm{t}} c : 1$. The inference of the transition

$$\Gamma_2 \triangleright P_2 \mid Q_2 \xrightarrow{\sigma} \sigma.R \mid Q_2$$

uses the rules (Sleep), (ActRcv) and (TimePar).

The final move we consider, $C_3 \twoheadrightarrow C_4 = \Gamma \triangleright R \mid \{^{w}/_{x}\}S$, is another instance of (Time). However here the delay action is inferred using (Sleep), (EndRcv) and (TimePar). Thus in three reduction

steps the value w has been transmitted from the first station to the second one along the channel c, in two units of time.

Now suppose we change P_1 to $P_1' = \sigma.P_1$, obtaining thus the configuration $C_1' = \Gamma_1 \triangleright P_1' \mid Q_1$. Then the first step, $C_1' \twoheadrightarrow C_2'$ is a (Time) step, with $C_2' = \Gamma_1 \triangleright P_1 \mid T_1$. Here an instance of the rule (Timeout) is used in the transition from Q_1 to T_1. In C_2' the station P_1 is now ready to transmit on channel c, but the second station has stopped listening. The next step depends on the exact form of T_1; if for example $\mathsf{rcv}(T_1, c)$ is false then by an application of rule (RcvIgn) we can derive $C_2' \twoheadrightarrow C_3' = \Gamma_2 \triangleright P_2 \mid T_1$. Here the transmission of w along c started but nobody was listening.

Finally, suppose T_1 is a delayed listener on channel c, say $\sigma.T_2$ where T_2 is $\lfloor c?(y).S_2 \rfloor U_2$. Then we have the (Time) step $C_3' \twoheadrightarrow C_4' = \Gamma_3 \triangleright \sigma.R \mid T_2$ and now the second station, T_2, is ready to listen. However, as $\Gamma_3 \vdash c : \mathbf{exp}$, station T_2 is joining the transmission too late. To reflect this we can derive we can derive the (Internal) step

$$C_4' \twoheadrightarrow C_5' = \Gamma_3 \triangleright \sigma.R \mid c[y].\{^{\mathsf{err}}/_y\}S_2$$

using the rules (RcvLate) and (TauPar), among others. At the end of the transmission, in one more time step, the second station will therefore end up with an error in reception.

In the revised system $C_1' = \Gamma_1 \triangleright \sigma.P_1' \mid Q_1$ the second station missed the delayed transmission from P_1'. However we can change the code at the second station to accommodate this delay, by replacing Q_1 with the persistent listener $Q_1' = c?(x).S$. We leave the reader to check that starting from the configuration $\Gamma_1 \triangleright \sigma.P_1' \mid Q_1'$ the value w will be successfully transmitted between the stations in four reduction steps. □

Example 2.17. [Collisions] Let us now consider again the configuration $C_1 = \Gamma \triangleright S_1 \mid S_2 \mid R_1$ of Example 2.1. In this configuration the station S_1 can perform a broadcast, leading to the reduction $C_1 \twoheadrightarrow C_2 = \Gamma_1 \triangleright \sigma^2 \mid S_2 \mid c[x].P$, the derivation of which requires an instance of the rule (RcvIgn), $\Gamma \triangleright S_1 \xrightarrow{c?v_1} S_1$; here the channel environment Γ_1 is defined as $\mathrm{upd}_{c!v_0}(\Gamma)$, leading to $\Gamma_1(c) = (2, v_0)$. We can now derive the reduction $C_2 \twoheadrightarrow C_3 = \Gamma_2 \triangleright \sigma \mid c!\langle v_1 \rangle \mid c[x].P$, where $\Gamma_2 = \mathrm{upd}_{\sigma}(\Gamma_1)$ meaning that $\Gamma_2 \vdash_{\mathrm{t}} c : 1$.

In this configuration the second station is ready to broadcast value v_1 along channel c. Since there is already a value being transmitted along this channel, we expect this second broadcast to cause a collision; further, since the amount of time required for transmitting value v_1 is equal to the time needed to end the transmission of value v_0, we expect that the broadcast performed by the first station does not affect the amount of time for which the channel c is exposed.

Formally this is reflected in the reduction $C_3 \twoheadrightarrow C_3' = \Gamma_2' \triangleright \sigma \mid \sigma \mid c[x].P$. Here the reduction of the system term uses the sub-inferences $\Gamma_2 \triangleright \sigma \xrightarrow{c?v_1} \sigma$, $\Gamma_2 \triangleright c!\langle v_1 \rangle \xrightarrow{c!v_1} \sigma$ and $\Gamma_2 \triangleright c[x].P \xrightarrow{c?v_1} c[x].P$; the first and the third of these transitions can be derived using Rule (RcvIgn), while the second one can be derived using Rule (Bcast). Consequently $\Gamma_2' = \mathrm{upd}_{c!v_1}(\Gamma_2)$, and since $\Gamma_2 \vdash c : \mathbf{exp}$ we obtain $\Gamma_2'(c) = (1, \mathsf{err})$; this represents the fact that a collision has occurred, and thus the special value err will eventually be delivered on c.

At this point we can derive the reductions $C_3' \twoheadrightarrow_{\sigma} C_4 = \Gamma \triangleright \mathsf{nil} \mid \mathsf{nil} \mid \{\mathsf{err}/x\}P$, meaning that the transmission along channel c terminates in one time instant, leading the receiving station to detect a collision. The reduction above can be obtained from the transitions $\Gamma_2' \triangleright \sigma \xrightarrow{\sigma} \mathsf{nil}$ and $\Gamma_2' \triangleright c[x].P \xrightarrow{\sigma} \{\mathsf{err}/x\}P$, obtained via rules (TimeNil) and (EndRcv) presented in Table 3.

Now, suppose we change the amount of time required to transmit value v_1 from 1 to 2, and consider again the configuration C_3 above. In this case the transmission of value v_1 will also cause a collision; however, in this case the transmission of value v_1 is long enough to continue after that of

value v_0 has finished; as a consequence, we expect that the time required for channel c to be released rises when the broadcast of v_1 happens.

In fact, in this case we have the reduction $C_3 \twoheadrightarrow C_3'' = \Gamma_2'' \triangleright \sigma \mid \sigma^2 \mid c[x].P$, where $\Gamma_2'' = \mathrm{upd}_{c!v_1}(\Gamma_2)$ and specifically $\Gamma_2''(c) = (2, \mathsf{err})$. Now, two time instants are needed for the transmission along channel c to end, leading to the sequence of (timed) reductions $C_3'' \twoheadrightarrow_\sigma \twoheadrightarrow_\sigma C_4$. □

2.4. **Behavioural semantics.** In this section we propose a notion of timed behavioural equivalence for our wireless networks. Our touchstone system equality is *reduction barbed congruence* [23, 46, 35, 25], a standard contextually defined process equivalence. Intuitively, two terms are reduction barbed congruent if they have the same *basic observables*, in all parallel contexts, under all possible *computations*. The formal definition relies on two crucial concepts, a reduction semantics to describe how systems evolve, which we have already defined, and a notion of basic observable which says what the environment can observe directly of a system. There is some choice as to what to take as a basic observation, or *barb*, of a wireless system. In standard process calculi this is usually taken to be the ability of the environment to receive a value along a channel. But the series of examples we have just seen demonstrates that this is problematic, in the presence of possible collisions and the passage of time. Instead we choose a more appropriate notion for wireless systems, one which is already present in our language for station code: *channel exposure*.

Definition 2.18. [Barbs] We say the configuration $\Gamma \triangleright W$ has a *strong barb on* c, written $\Gamma \triangleright W \downarrow_c$, if $\Gamma \vdash c : \mathbf{exp}$. We write $\Gamma \triangleright W \Downarrow_c$, a *weak barb*, if there exists a configuration C' such that $\Gamma \triangleright W \twoheadrightarrow^* C'$ and $C' \downarrow_c$. Note that we allow the passage of time in the definition of weak barb. □

Definition 2.19. Let $\mathcal{R}$ be a relation over configurations.

(1) $\mathcal{R}$ is said to be *barb preserving* if $\Gamma_1 \triangleright W_1 \Downarrow_c$ implies $\Gamma_2 \triangleright W_2 \Downarrow_c$, whenever $(\Gamma_1 \triangleright W_1)\ \mathcal{R}\ (\Gamma_2 \triangleright W_2)$.
(2) It is *reduction-closed* if $(\Gamma_1 \triangleright W_1)\ \mathcal{R}\ (\Gamma_2 \triangleright W_2)$ and $\Gamma_1 \triangleright W_1 \twoheadrightarrow \Gamma_1' \triangleright W_1'$ imply there is some $\Gamma_2' \triangleright W_2'$ such that $\Gamma_2 \triangleright W_2 \twoheadrightarrow^* \Gamma_2' \triangleright W_2'$ and $(\Gamma_1' \triangleright W_1')\ \mathcal{R}\ (\Gamma_2' \triangleright W_2')$.
(3) It is *contextual* if $\Gamma_1 \triangleright W_1\ \mathcal{R}\ \Gamma_2 \triangleright W_2$, implies $\Gamma_1 \triangleright (W_1 \mid W)\ \mathcal{R}\ \Gamma_2 \triangleright (W_2 \mid W)$ for all processes W.

□

With these concepts we now have everything in place for a standard definition of contextual equivalence between systems:

Definition 2.20. [Reduction barbed congruence], written $\simeq$, is the largest symmetric relation over configurations which is barb preserving, reduction-closed and contextual. □

In the remainder of this section we explore via examples the implications of Definition 2.20. The notion of a fresh channel will be important; we say that c is *fresh* for the configuration $\Gamma \triangleright W$ if it does not occur free in W and $\Gamma \vdash c : \mathbf{idle}$. Note that we can always pick a fresh channel for an arbitrary configuration.

Example 2.21. Let us assume that $\Gamma \vdash c : \mathbf{idle}$. Then it is easy to see that

$$\Gamma \triangleright c\,!\langle v_0\rangle.P \not\simeq \Gamma \triangleright c\,!\langle v_1\rangle.P \tag{2.4}$$

under the assumption that v_0 and v_1 are different values. For let T be the testing context

$$\lfloor c?(x).[x = v_0]\mathit{eureka}!\langle \mathit{ok}\rangle, \mathsf{nil}\rfloor$$

where *eureka* is fresh, and *ok* is some arbitrary value. Then $\Gamma \triangleright c\,!\langle v_0\rangle.P \mid T$ has a weak barb on *eureka* which is not the case for $\Gamma \triangleright c\,!\langle v_1\rangle.P \mid T$. Since $\simeq$ is contextual and barb preserving, the statement (2.4) above follows.

However such tests will not distinguish between $\Gamma \triangleright Q_1$ and $\Gamma \triangleright Q_2$, where

$$Q_1 = c!\langle v_0\rangle \mid c\,!\langle v_1\rangle.P \qquad \text{and} \qquad Q_2 = c!\langle v_1\rangle \mid c\,!\langle v_0\rangle.P$$

assuming that $\delta_{v_0} = \delta_{v_1}$. In both configurations $\Gamma \triangleright Q_1$ and $\Gamma \triangleright Q_2$ a collision will occur at channel c and a receiving station, such as T, will receive the error value err at the end of the transmission. So there is reason to hope that $\Gamma \triangleright Q_1 \simeq \Gamma \triangleright Q_2$. However we must wait for for the proof techniques of the next section to establish this equivalence; see Example 3.5. □

The above example suggests that transmitted values can be observed only at the end of a transmission; so if a collision happens, there is no possibility of determining the value that was originally broadcast. This concept is stressed even more in the following example.

Example 2.22. [Equating values] Let Γ be a stable channel environment, $W_0 = c!\langle v_0\rangle, W_1 = c!\langle v_1\rangle$ and consider the configurations $\Gamma \triangleright W_0, \Gamma \triangleright W_1$; here we assume that v_0 and v_1 are two different values with possibly different transmission times.

We already argued in Example 2.21 that these two configurations can be distinguished by the context

$$\lfloor c?(x).[x = v_0]\mathit{eureka}!\langle \mathit{ok}\rangle, \mathsf{nil}\rfloor$$

However, the two configurations above can be made indistinguishable if we add to each of them a parallel component that causes a collision on channel c. To this end, let

$$Eq(v_0, v_1) = \sigma^h.c!\langle \mathit{ok}\rangle$$

for some positive integer h and value *ok* such that $h < \min(\delta_{v_0}, \delta_{v_1})$ and $\delta_{ok} \geq \max(\delta_{v_0}, \delta_{v_1}) - h$. Now, consider the configurations $C_0 = \Gamma \triangleright W_0 \mid Eq(v_0, v_1), C_1 = \Gamma \triangleright W_1 \mid Eq(v_0, v_1)$.

One could hope that there exists a context which is able to distinguish these two configurations. However, before the transmission of v_0 ends in C_0, a second broadcast along channel c will fire, causing a collision; the same happens before the end of transmission of value v_1 in C_1. Further, the total amount of time for which channel c will be exposed is the same for both configurations, so that one can argue that it is impossible to provide a context which is able to distinguish C_0 from C_1. In order to prove this to be formally true, we have to wait until the next section. □

Collisions can also be used to merge two different transmissions on the same channel in a single corrupted transmission.

Example 2.23. [Merging Transmissions] Let Γ be a stable channel environment, $W_0 = c\,!\langle v_0\rangle.c!\langle v_1\rangle$, $W_1 = c\,!\langle v_1\rangle.c!\langle v_0\rangle$. In $\Gamma \triangleright W_0$ a broadcast of value v_0 along channel c can fire; when the transmission of v_0 is finished, a second broadcast of value v_1 along the same channel can also fire. The behaviour of $\Gamma \triangleright W_1$ is similar, though the order of the two values to be broadcast is swapped. Note that it is possible to distinguish the two configurations $\Gamma \triangleright W_0$ and $\Gamma \triangleright W_1$ using the test

$$\lfloor c?(x).[x = v_0]\mathit{eureka}!\langle \mathit{ok}\rangle, \mathsf{nil}\rfloor$$

we have already seen in the previous example.

However suppose now that we add a parallel component to both configurations which broadcasts another value along channel c before the transmission of value v_0 (v_1) has finished, and which terminates after the broadcast of value v_1 (v_0) has begun. More formally, let

$$Mrg(v_0, v_1) = \sigma^h.c!\langle \mathit{ok}\rangle$$

where $h = \min(\delta_{v_0}, \delta_{v_1}) - 1$ and $\delta_{ok} = |\delta_{v_0} - \delta_{v_1}| + 2$.

Consider the configurations $\Gamma \triangleright W_0 \mid Mrg(v_0, v_1)$, $\Gamma \triangleright W_1 \mid Mrg(v_0, v_1)$. In both configurations a collision occurs; further, once the transmission of value v_0 has begun in the former configuration, channel c will remain exposed until the transmission of value v_1 has finished. A similar behaviour can be observed on the second configuration. This leads to the intuition that $\Gamma \triangleright W_0 \mid Mrg(v_0, v_1) \simeq \Gamma' \triangleright W_1 \mid Mrg(v_0, v_1)$; we prove this in Example 3.7, for a particular instance of transmission values for v_0, v_1. □

A priori reductions ignore the passage of time, and therefore one might suspect that reduction barbed congruence is impervious to the precise timing of activities. But the next example demonstrates that this is not the case.

Example 2.24. [Observing the passage of time] Consider the two processes $Q_1 = c!\langle v_0\rangle$ and $Q_2 = \sigma.Q_1$, and again let us assume that $\Gamma \vdash c : \mathbf{idle}$. There is very little difference between the behaviours of $\Gamma \triangleright Q_1$ and $\Gamma \triangleright Q_2$; both will transmit (successfully) the value v_0, although the latter is a little slower. However this slight difference can be observed. Consider the test T defined by

$$[\exp(c)]\mathit{eureka}!\langle ok\rangle, \mathsf{nil}$$

In fact, $\Gamma \triangleright (Q_1 \mid T)$ can start a transmission along channel c, after which the predicate $\exp(c)$ will be evaluated in the system term T. The resulting configuration is given by $\Gamma' \triangleright \sigma^{\delta_{v_0}} \mid \sigma.\mathit{eureka}!\langle ok\rangle$; at this point, it is not difficult to note that the configuration has a weak barb on *eureka*.

On the other hand, the *unique* reduction from $C_2 = \Gamma \triangleright (Q_2 \mid T)$ leads to the evaluation of the exposure predicate $\exp(c)$; since $\Gamma \vdash c : \mathbf{idle}$ the only possibility for the resulting configuration is given by $C_2' = \Gamma \triangleright Q_2 \mid \sigma$. Since *eureka* is a fresh channel, it is now immediate to note that $C_2' \Downarrow_{\mathit{eureka}}$ and hence also $C_2 \Downarrow_{\mathit{eureka}}$. For the test to work correctly it is essential that $\Gamma \vdash c : \mathbf{idle}$. Here we would like to point out that using the proof methodology developed in Section 3.2 we are able to show that if $\Gamma' \vdash_{\mathrm{t}} c : n$ and $n > \delta_{v_0}$ then $\Gamma' \triangleright Q_1 \simeq \Gamma' \triangleright Q_2$. □

Behind this example is the general principle that reduction barbed congruence is actually sensitive to the passage of time; this is proved formally in Proposition 4.17 of Section 4.2.

Example 2.25. As a final example we illustrate the use of channel restriction. Assume that v_1 and v_2 are some kind of values which can be compared via a (total) order relation $\leqslant$. Consider the configuration
$\Gamma \triangleright \nu c : (0, \cdot).(c!\langle v_1\rangle \mid P_e \mid R)$ where the station code is given by

$$\begin{aligned} P_e &= \sigma.\mathsf{fix}\, X.([\exp(c)]X, c!\langle v_2\rangle) \\ R &= c?(x).R_1 \\ R_1 &= c?(y).[y \leqslant x]d!\langle x\rangle, d!\langle y\rangle \end{aligned}$$

Intuitively the receiver R waits indefinitely for two values along the restricted channel c and broadcasts the largest on channel d. Intuitively the use of channel restriction here shelters c from external interference. Assuming $\Gamma \vdash d : \mathbf{idle}$ we will be able to show that

$$\Gamma \triangleright \nu c : (0, \cdot).(c\,!\langle v_1\rangle.\mathsf{nil} \mid P_e \mid R) \quad \simeq \quad \Gamma \triangleright \sigma^{\delta_{v_1} + \delta_{v_2} + 2}.d\,!\langle w\rangle.\mathsf{nil}$$

provided $w = \max(v_1, v_2)$. □

Table 6 Extensional actions

$$\text{(Input)}\ \frac{\Gamma \triangleright W \xrightarrow{c?v} W'}{\Gamma \triangleright W \overset{c?v}{\longmapsto} \mathrm{upd}_{c?v}(\Gamma) \triangleright W'} \qquad \text{(Time)}\ \frac{\Gamma \triangleright W \xrightarrow{\sigma} W'}{\Gamma \triangleright W \overset{\sigma}{\longmapsto} \mathrm{upd}_{\sigma}(\Gamma) \triangleright W'}$$

$$\text{(Shh)}\ \frac{\Gamma \triangleright W \xrightarrow{c!v} W'}{\Gamma \triangleright W \overset{\tau}{\longmapsto} \mathrm{upd}_{c!v}(\Gamma) \triangleright W'} \qquad \text{(TauExt)}\ \frac{\Gamma \triangleright W \xrightarrow{\tau} W'}{\Gamma \triangleright W \overset{\tau}{\longmapsto} \Gamma \triangleright W'}$$

$$\text{(Deliver)}\ \frac{\Gamma(c) = (1, v) \quad \Gamma \triangleright W \xrightarrow{\sigma} W'}{\Gamma \triangleright W \overset{\gamma(c,v)}{\longmapsto} \mathrm{upd}_{\sigma}(\Gamma) \triangleright W'} \qquad \text{(Idle)}\ \frac{\Gamma \vdash c : \mathbf{idle}}{\Gamma \triangleright W \overset{\iota(c)}{\longmapsto} \Gamma \triangleright W}$$

3. Extensional Semantics

Proving that two configurations C_1 and C_2 are barbed congruent can be difficult, due to the contextuality constraint imposed in Definition 2.20. Therefore, we want to give a co-inductive characterisation of the contextual equivalence $\simeq$ between configurations, in terms of a standard bisimulation equivalence over some extensional LTS. In this section we first present the extensional semantics, then we recall the standard definition of (weak) bisimulation over configurations. We show, by means of a number of examples, the usefulness of the actions introduced in the extensional semantics.

3.1. **Extensional actions.** The extensional semantics is designed by addressing the question: what actions can be detected by an external observer? Example 2.24 indicates that the passage of time is observable. The effect of inputs received from the external environment also has to be taken into account. In contrast, the discussion in Example 2.21 indicates that, due to the possibility of collisions, the treatment of transmissions is more subtle. It turns out that the transmission itself is not important; instead we must take into consideration the successful delivery of the transmitted value.

In Table 6 we give the rules defining the extensional actions, $C \overset{\alpha}{\longmapsto} C'$, which can take one of the forms:

- Input: $C \overset{c?v}{\longmapsto} C'$, this is inherited directly from the intensional semantics
- Time: $C \overset{\sigma}{\longmapsto} C'$, also inherited from the intensional semantics
- Internal: $C \overset{\tau}{\longmapsto} C'$, this corresponds to the combination of the Internal and Transmission rules from the reduction semantics, in Definition 2.14
- Delivery: $C \overset{\gamma(c,v)}{\longmapsto} C'$, this corresponds to the successful delivery of the value v which was in transmission along the channel c
- Free: $C \overset{\iota(c)}{\longmapsto} C$, a predicate indicating that channel c is not exposed, and therefore ready to start a potentially successful transmission.

Remark 3.1. The rules provided in Table 6 guarantee that τ-extensional actions coincide with instantaneous reductions. In fact, whenever $\Gamma \triangleright W \twoheadrightarrow \Gamma' \triangleright W'$ then either $\Gamma \triangleright W \xrightarrow{\tau} W'$, and hence $\Gamma \triangleright W \overset{\tau}{\longmapsto} \Gamma' \triangleright W'$ follows by an application of Rule (ExtTau), with $\Gamma' = \mathrm{upd}_{\tau}(\Gamma)$, or $\Gamma \triangleright W \xrightarrow{c!v} W'$

and $\Gamma \triangleright W \overset{\tau}{\longmapsto} \Gamma' \triangleright W'$ is ensured by Rule (Shh), with $\Gamma' = \mathrm{upd}_{c!v}(\Gamma)$. The opposite implication can be proved analogously.

Similarly, it is easy to check extensional σ-actions coincide with timed reductions: $\Gamma \triangleright W \twoheadrightarrow_\sigma \Gamma' \triangleright W'$ if and only if $\Gamma \triangleright W \overset{\sigma}{\longmapsto} \Gamma' \triangleright W'$. □

3.2. **Bisimulation equivalence.** The extensional actions of the previous section endows systems in CCCP with the structure of an LTS. Weak extensional actions in this LTS are defined as usual, with $C \overset{\alpha}{\Longmapsto} C'$ denoting $C \overset{\tau}{\longmapsto}{}^* \overset{\alpha}{\longmapsto} \overset{\tau}{\longmapsto}{}^* C'$. We will use $C \Longmapsto C'$ to denote $C \overset{\tau}{\longmapsto}{}^* C'$, and the formulation of bisimulations is facilitated by the notation $C \overset{\hat{\alpha}}{\Longmapsto} C'$, which is again standard: for $\alpha = \tau$ this denotes $C \Longmapsto C'$ while for $\alpha \neq \tau$ it is $C \overset{\alpha}{\Longmapsto} C'$. We now have the standard definition of weak bisimulation equivalence in the resulting LTS which for convenience we recall.

Definition 3.2. Let $\mathcal{R}$ be a binary relation over configurations. We say that $\mathcal{R}$ is a bisimulation if for every extensional action α, whenever $C_1 \mathcal{R} C_2$

(i) $C_1 \overset{\alpha}{\longmapsto} C_1'$ implies $C_2 \overset{\hat{\alpha}}{\Longmapsto} C_2'$, for some C_2', satisfying $C_1' \mathcal{R} C_2'$

(ii) conversely, $C_2 \overset{\alpha}{\longmapsto} C_2'$ implies $C_1 \overset{\hat{\alpha}}{\Longmapsto} C_1'$, for some C_1', such that $C_1' \mathcal{R} C_2'$.

We write $C_1 \approx C_2$, if there is a bisimulation $\mathcal{R}$ such that $C_1 \mathcal{R} C_2$. □

Our goal is to demonstrate that this form of bisimulation provides a sound and useful proof method for showing behavioural equivalence between wireless systems described in CCCP; moreover for a large class of systems it will also turn out to be complete.

The next two examples show that the introduction of the actions $\iota(c)$ and $\gamma(c, v)$ are necessary for soundness.

Example 3.3. [On the rule (Idle)] Suppose we were to drop the rule (Idle) in the extensional semantics; then consider the configurations

$$\begin{aligned} \Gamma_1 \triangleright W_1 &= \tau.\mathsf{nil} \\ \Gamma_2 \triangleright W_2 &= c!\langle v \rangle \end{aligned}$$

where $\Gamma_1(c) = (1, v)$, $\Gamma_2(c) = (0, \cdot)$ and $\delta_v = 1$.

If we were to drop the actions $\iota(c)$ from the extensional semantics then the extensional LTSs generated by these two configurations would be isomorphic; recall that a broadcast action in the intensional semantics always corresponds to a τ action in its extensional counterpart. Thus they would be related by the amended version of bisimulation equivalence.

However, we also have that $\Gamma_1 \triangleright W_1 \not\approx \Gamma_2 \triangleright W_2$. This can be proved by exhibiting a distinguishing context. To this end, consider the system $T = [\exp(c)]\mathsf{nil}, \mathrm{eureka}!\langle ok \rangle$. Then $\Gamma_2 \triangleright W_2 \mid T$ has a weak barb on the channel eureka, which obviously $\Gamma_2 \triangleright W_1 \mid T$ can not match. □

Example 3.4. [On the rule (Deliver)] Consider the configuration $\Gamma_2 \triangleright W_2$ from the previous example; consider also the configuration $\Gamma_2 \triangleright W_3$, where $W_3 = c!\langle w \rangle$ for some value w, different from v, such that $\delta_w = 1$. Finally, let $T' = \lfloor c?(x).[x = v]\mathrm{eureka}!\langle ok \rangle, \mathsf{nil} \rfloor$. Then, assuming w is different from v, $\Gamma_2 \triangleright W_3 \mid T'$ can not produce a barb on eureka. On the other hand, $\Gamma_2 \triangleright W_2 \mid T'$ can produce such a barb. It follows that $\Gamma_2 \triangleright W_2 \not\approx \Gamma_2 \triangleright W_3$.

Note also that $\Gamma_2 \triangleright W_3 \not\approx \Gamma_2 \triangleright W_2$, since the (weak) action $\Gamma_2 \triangleright W_3 \overset{\gamma(c,w)}{\Longmapsto} \Gamma \triangleright \mathsf{nil}$ cannot be matched by $\Gamma_2 \triangleright W_2$. However, if we were to drop the rule (Deliver) in the extensional semantics, thereby eliminating the actions $\gamma(c, v)$, then it would be straightforward to exhibit a bisimulation containing

the pair $(\Gamma_2 \triangleright W_3, \Gamma_2 \triangleright W_2)$. Thus again the amended version of bisimulation equivalence would be unsound. □

The two examples above show that both rules (Idle) and (Deliver) are necessary to achieve the soundness of our bisimulation proof method for reduction barbed congruence.

In the remainder of this section we give a further series of examples, showing that bisimulations in our extensional LTS offer a viable proof technique for demonstrating behavioural equivalence for at least simple wireless systems.

Example 3.5. [Transmission] Here we revisit Example 2.21. Let Γ be a stable channel environment, and consider the configurations $C_0 = \Gamma \triangleright W$, $C_1 = \Gamma \triangleright V$, where $W = c\,!\langle v_0\rangle.P \mid c!\langle v_1\rangle$, $V = c\,!\langle v_1\rangle.P \mid c!\langle v_0\rangle$; note that these two configurations are taken from the second part of Example 2.21.

Our aim is to show that $C_0 \approx C_1$, when $\delta_{v_0} = \delta_{v_1}$; for convenience let us assume that $\delta_{v_0} = \delta_{v_1} = 1$. The idea here is to describe the required bisimulation by matching up system terms. To this end we define the following system terms:

$$\begin{array}{rclcrcl} W_0 & = & \sigma.P \mid c!\langle v_1\rangle & \qquad & V_1 & = & \sigma.P \mid c!\langle v_0\rangle \\ W_1 & = & c\,!\langle v_0\rangle.P \mid \sigma & & V_0 & = & c\,!\langle v_1\rangle.P \mid \sigma \\ E & = & \sigma.P \mid \sigma & & E' & = & P \mid \mathsf{nil} \end{array}$$

Then for any channel environment Δ we have the following transitions in the extensional semantics:

$$\begin{array}{lcl@{\qquad}lcl} \Delta \triangleright W & \overset{\tau}{\longmapsto} & \mathrm{upd}_{c!v_0}(\Delta) \triangleright W_0 & \Delta \triangleright V & \overset{\tau}{\longmapsto} & \mathrm{upd}_{c!v_0}(\Delta) \triangleright V_0 \\ \Delta \triangleright W & \overset{\tau}{\longmapsto} & \mathrm{upd}_{c!v_1}(\Delta) \triangleright W_1 & \Delta \triangleright V & \overset{\tau}{\longmapsto} & \mathrm{upd}_{c!v_1}(\Delta) \triangleright V_1 \\ \Delta \triangleright W & \overset{d?w}{\longmapsto} & \mathrm{upd}_{d?w}(\Delta) \triangleright W & \Delta \triangleright V & \overset{d?w}{\longmapsto} & \mathrm{upd}_{d?w}(\Delta) \triangleright V \\ \Delta \triangleright W & \overset{\iota(d)}{\longmapsto} & \Delta \triangleright W \text{ if } \Delta \vdash d : \mathbf{idle} & \Delta \triangleright V & \overset{\iota(d)}{\longmapsto} & \Delta \triangleright V \text{ if } \Delta \vdash d : \mathbf{idle} \end{array}$$

$$\begin{array}{lcl@{\qquad}lcl} \Delta \triangleright W_0 & \overset{\tau}{\longmapsto} & \mathrm{upd}_{c!v_1}(\Delta) \triangleright E & \Delta \triangleright V_0 & \overset{\tau}{\longmapsto} & \mathrm{upd}_{c!v_1}(\Delta) \triangleright E \\ \Delta \triangleright W_0 & \overset{d?w}{\longmapsto} & \mathrm{upd}_{d?w}(\Delta) \triangleright W_0 & \Delta \triangleright V_0 & \overset{d?w}{\longmapsto} & \mathrm{upd}_{d?w}(\Delta) \triangleright V_0 \\ \Delta \triangleright W_0 & \overset{\iota(d)}{\longmapsto} & \Delta \triangleright W_0 \text{ if } \Delta \vdash d : \mathbf{idle} & \Delta \triangleright V_0 & \overset{\iota(d)}{\longmapsto} & \Delta \triangleright V_0 \text{ if } \Delta \vdash d : \mathbf{idle} \end{array}$$

$$\begin{array}{lcl@{\qquad}lcl} \Delta \triangleright W_1 & \overset{\tau}{\longmapsto} & \mathrm{upd}_{c!v_0}(\Delta) \triangleright E & \Delta \triangleright V_1 & \overset{\tau}{\longmapsto} & \mathrm{upd}_{c!v_0}(\Delta) \triangleright E \\ \Delta \triangleright W_1 & \overset{d?w}{\longmapsto} & \mathrm{upd}_{d?w}(\Delta) \triangleright W_1 & \Delta \triangleright V_1 & \overset{d?w}{\longmapsto} & \mathrm{upd}_{d?w}(\Delta) \triangleright V_1 \\ \Delta \triangleright W_1 & \overset{\iota(d)}{\longmapsto} & \Delta \triangleright W_1 \text{ if } \Delta \vdash d : \mathbf{idle} & \Delta \triangleright V_1 & \overset{\iota(d)}{\longmapsto} & \Delta \triangleright V_1 \text{ if } \Delta \vdash d : \mathbf{idle} \end{array}$$

Here d ranges over arbitrary channel names, including c.

Then consider the following relation:

$$\mathcal{S} = \{(\Delta \triangleright W, \Delta \triangleright V),\ (\Delta \triangleright W_0,\ \Delta \triangleright V_0), (\Delta \triangleright W_1, \Delta \triangleright V_1) \mid \Delta \text{ is a channel environment}\}\ .$$

Using the above tabulation of actions one can now show that $\mathcal{S}$ is a *strong* bisimulation; for $C\,\mathcal{S}\,C'$ each possible action of C can be matched by C' by performing exactly the same action, and vice-versa.

Since $(C_0, C_1) \in \mathcal{S}$, it follows that $C_0 \approx C_1$. □

Table 7 A relation $\mathcal{S}$ for comparing the configurations C_0, C_1 of Example 3.6

$$
\begin{array}{rcl}
\Delta \triangleright W & \mathcal{S} & \Delta \triangleright V \\
\Delta \triangleright W_0 & \mathcal{S} & \Delta \triangleright V_0 \\
(\Delta[c \mapsto (1, v_0)]) \triangleright W_0 & \mathcal{S} & (\Delta[c \mapsto (2, v_1)]) \triangleright V_1 \\
(\Delta[c \mapsto (1, \mathsf{err})]) \triangleright W_0 & \mathcal{S} & (\Delta[c \mapsto (2, \mathsf{err})]) \triangleright V_1 \\
\Lambda \triangleright W_{ok} & \mathcal{S} & \Lambda \triangleright V_{ok} \\
\Delta \triangleright W_{\mathsf{err}} & \mathcal{S} & \Delta \triangleright V_{\mathsf{err}} \\
\Delta \triangleright W' & \mathcal{S} & \Delta \triangleright V'
\end{array}
$$

Δ arbitrary channel environment,
Λ arbitrary channel environment such that $\Lambda(c) = (k, w)$ for some $k \geq 2$

Example 3.6. [Equators] Let us consider the configurations C_0, C_1 of Example 2.22. Recall that $C_0 = \Gamma \triangleright W$, where $W = c!\langle v_0 \rangle \mid \sigma^h.c!\langle ok \rangle$ and $C_1 = \Gamma \triangleright V$, where $V = c!\langle v_1 \rangle \mid \sigma^h.c!\langle ok \rangle$; further, recall that Γ is a stable channel environment and h, ok are a positive integer and a value, respectively, such that $h < \min(\delta_{v_0}, \delta_{v_1})$, $\delta_{ok} \geq \max(\delta_{v_0}, \delta_{v_1}) - h$. Without loss of generality, for this example we assume $\delta_{v_0} = 1, \delta_{v_1} = 2$, $h = 0$ and $\delta_{ok} = 2$.

For the sake of convenience we define the following system terms:

$$
\begin{array}{rclcrcl}
W_0 & = & \sigma \mid c!\langle ok \rangle & & V_1 & = & \sigma^2 \mid c!\langle ok \rangle \\
W_{ok} & = & c!\langle v_0 \rangle \mid \sigma^2 & & V_{ok} & = & c!\langle v_1 \rangle \mid \sigma^2 \\
W_{\mathsf{err}} & = & \sigma \mid \sigma^2 & & V_{\mathsf{err}} & = & \sigma^2 \mid \sigma^2 \\
W' & = & \mathsf{nil} \mid \sigma & & V' & = & \sigma \mid \sigma \\
E & = & \mathsf{nil} \mid \mathsf{nil} & & & &
\end{array}
$$

Let us consider the relation $\mathcal{S}$ depicted in Table 7; note that $(C_0, C_1) \in \mathcal{S}$, so that in order to prove that $C_0 \approx C_1$ it is sufficient to show that $\mathcal{S}$ is a bisimulation. Note that in the relation $\mathcal{S}$ the system terms W_{ok}, V_{ok} are always associated with a channel environment in which the channel c is exposed. In fact, if Λ were a channel environment such that $\Lambda \vdash c : \mathbf{idle}$, it would not be difficult to prove that $\Lambda \triangleright W_{\mathsf{err}} \not\approx \Lambda \triangleright V_{\mathsf{err}}$; this is because the values broadcast by these two configurations are different.

Table 8 A relation $\mathcal{S}$ for comparing the configurations C_0, C_1 of Example 3.7

$$
\begin{array}{rcl}
\Delta \triangleright W & \mathcal{S} & \Delta \triangleright V \\
\Delta[c \mapsto (1,w)] \triangleright W_0 & \mathcal{S} & \Delta[c \mapsto (2,w)] \triangleright V_1 \\
\Delta[c \mapsto (k+2,w) \triangleright W_0 & \mathcal{S} & \Delta[c \mapsto (k+2,w)] \triangleright V_1 \\
\Delta[c \mapsto (k+3,w)] W_{ok} \triangleright & \mathcal{S} & \Delta[c \mapsto (k+3,w)] \triangleright V_{ok} \\
\Delta[c \mapsto (k+3,\mathsf{err}) \triangleright W_{\mathsf{err}} & \mathcal{S} & \Delta[c \mapsto (k+3,\mathsf{err})] \triangleright V_{\mathsf{err}} \\
\Delta[c \mapsto (k+2,\mathsf{err}) \triangleright W' & \mathcal{S} & \Delta[c \mapsto (k+2,\mathsf{err}) \triangleright V' \\
\Delta[c \mapsto (k+2,\mathsf{err})] \triangleright W_1 & \mathcal{S} & \Delta[c \mapsto (k+2,\mathsf{err})] \triangleright V' \\
\Delta[c \mapsto (k+1,\mathsf{err})] \triangleright E' & \mathcal{S} & \Delta[c \mapsto (k+1,\mathsf{err})] \triangleright V''
\end{array}
$$

Δ arbitrary channel environment, w arbitrary value (possibly err) and $k \geq 0$.

Let us list the main the extensional actions from configurations using these system terms:

$$
\begin{array}{lcl}
\Delta \triangleright W & \stackrel{\tau}{\longmapsto} & (\Delta[c \mapsto (1,v_0)]) \triangleright W_0 \text{ if } \Delta \vdash c : \mathbf{idle} \\
\Delta \triangleright V & \stackrel{\tau}{\longmapsto} & (\Delta[c \mapsto (2,v_1)]) \triangleright V_1 \text{ if } \Delta \vdash c : \mathbf{idle} \\
\Delta \triangleright W & \stackrel{\tau}{\longmapsto} & (\Delta[c \mapsto (2,ok)]) \triangleright W_{ok} \\
\Delta \triangleright V & \stackrel{\tau}{\longmapsto} & (\Delta[c \mapsto (2,ok)]) \triangleright V_{ok} \\
\Delta \triangleright W & \stackrel{d?w}{\longmapsto} & (\mathrm{upd}_{d?w}(\Delta)) \triangleright W \\
\Delta \triangleright V & \stackrel{d?w}{\longmapsto} & (\mathrm{upd}_{d?w}(\Delta)) \triangleright V \\
 & & \\
(\Delta[c \mapsto (1,v_0)]) \triangleright W_0 & \stackrel{\tau}{\longmapsto} & (\Delta[c \mapsto (2,\mathsf{err})]) \triangleright W_{\mathsf{err}} \\
(\Delta[c \mapsto (2,v_1)]) \triangleright V_1 & \stackrel{\tau}{\longmapsto} & (\Delta[c \mapsto (2,\mathsf{err})]) \triangleright W_{\mathsf{err}} \\
\Delta \triangleright W_0 & \stackrel{c?w}{\longmapsto} & (\Delta[c \mapsto (1,\mathsf{err})]) \triangleright W_0 \text{ if } \Delta \vdash c : \mathbf{exp}, \delta_w = 1 \\
\Delta \triangleright V_1 & \stackrel{c?w}{\longmapsto} & (\Delta[c \mapsto (2,\mathsf{err})]) \triangleright V_1 \text{ if } \Delta \vdash c : \mathbf{exp}, \delta_w = 1 \\
\Delta \triangleright W_0 & \stackrel{c?w}{\longmapsto} & (\Delta[c \mapsto (\delta_w,\mathsf{err})]) \triangleright W_0 \text{ if } \Delta \vdash c : \mathbf{exp}, \delta_w > 1 \\
\Delta \triangleright V_1 & \stackrel{c?w}{\longmapsto} & (\Delta[c \mapsto (\delta_w,\mathsf{err})]) \triangleright V_1 \text{ if } \Delta \vdash c : \mathbf{exp}, \delta_w > 1 \\
 & & \\
\Lambda \triangleright W_{ok} & \stackrel{\tau}{\longmapsto} & (\mathrm{upd}_{c!v_0}(\Lambda)) \triangleright W_{\mathsf{err}} \\
\Lambda \triangleright V_{ok} & \stackrel{\tau}{\longmapsto} & (\mathrm{upd}_{c!v_1}(\Lambda)) \triangleright V_{\mathsf{err}} \\
 & & \\
\Delta \triangleright W_{\mathsf{err}} & \stackrel{\sigma}{\longmapsto} & (\mathrm{upd}_{\sigma}(\Delta)) \triangleright W' \\
\Delta \triangleright V_{\mathsf{err}} & \stackrel{\sigma}{\longmapsto} & (\mathrm{upd}_{\sigma}(\Delta)) \triangleright V' \\
 & & \\
\Delta \triangleright W' & \stackrel{\sigma}{\longmapsto} & (\mathrm{upd}_{\sigma}(\Delta)) \triangleright E \\
\Delta \triangleright V' & \stackrel{\sigma}{\longmapsto} & (\mathrm{upd}_{\sigma}(\Delta)) \triangleright E
\end{array}
$$

Here Δ, Λ are two arbitrary channel environments, but Λ is subject to the constraint that $\Lambda(c) = (k,w)$ for some value w and integer $k \geq 2$. This last requirement ensures that $(\mathrm{upd}_{c!v_0}(\Lambda)) = (\mathrm{upd}_{c!v_1}(\Lambda))$. With the aid of this tabulation one can now show that $\mathcal{S}$ is indeed a bisimulation and therefore that $C_0 \approx C_1$. □

Example 3.7. [Merging] The last example we provide considers the merging of two transmissions in a single transmission as suggested in the Example 2.23. Let Γ be a stable channel environment

and v_0, v_1 be two values such that $\delta_{v_0} = 1, \delta_{v_1} = 2$. Also let ok be a value such that $\delta_{ok} = 3$. Consider the configurations

$$C_0 = \Gamma \triangleright W \qquad\qquad C_1 = \Gamma \triangleright V$$

where $W = c\,!\langle v_0\rangle.c!\langle v_1\rangle \mid c!\langle ok\rangle$ and $V = c\,!\langle v_1\rangle.c!\langle v_0\rangle \mid c!\langle ok\rangle$.

Then $C_0 \approx C_1$. As in previous examples, this statement can be proved formally by exhibiting a bisimulation that contains the pair (C_0, C_1); to this end, define the following system terms:

$$\begin{array}{rclrcl}
W_0 & = & \sigma.c!\langle v_1\rangle \mid c!\langle ok\rangle & V_1 & = & \sigma^2.c!\langle v_0\rangle \mid c!\langle ok\rangle \\
W_{ok} & = & c\,!\langle v_0\rangle.c!\langle v_1\rangle \mid \sigma^3 & V_{ok} & = & c\,!\langle v_1\rangle.c!\langle v_0\rangle \mid \sigma^3 \\
W_{\mathsf{err}} & = & \sigma.c!\langle v_1\rangle \mid \sigma^3 & W_{\mathsf{err}} & = & \sigma^2.c!\langle v_0\rangle \mid \sigma^3 \\
W' & = & c!\langle v_1\rangle \mid \sigma^2 & & & \\
W_1 & = & \sigma^2 \mid \sigma^2 & V' & = & \sigma.c!\langle v_0\rangle \mid \sigma^2 \\
E' & = & \sigma \mid \sigma & V'' & = & c!\langle v_0\rangle \mid \sigma \\
E & = & \mathsf{nil} \mid \mathsf{nil} & & &
\end{array}$$

Consider now the relation $\mathcal{S}$ depicted in Table 8; note that $C_0\ \mathcal{S}\ C_1$. Also, $\mathcal{S}$ is a weak bisimulation. In order to show this, we list the non-trivial transitions for both configurations C_0, C_1 and their derivatives, which are needed to perform the proof.

$$\begin{array}{lcl}
\Delta[(c \mapsto (0,\cdot)] \triangleright W & \overset{\tau}{\longmapsto} & \Delta[c \mapsto (1, v_0)] \triangleright W_0 \\
\Delta[(c \mapsto (0,\cdot)] \triangleright V & \overset{\tau}{\longmapsto} & \Delta[c \mapsto (2, v_1)] \triangleright V_1 \\
 & & \\
\Delta[c \mapsto (0,\cdot)] \triangleright W & \overset{\tau}{\longmapsto} & \Delta[c \mapsto (3, ok)] \triangleright W_{ok} \\
\Delta[c \mapsto (0,\cdot)] \triangleright V & \overset{\tau}{\longmapsto} & \Delta[c \mapsto (3, ok])] \triangleright V_{ok} \\
 & & \\
\Delta[c \mapsto (k,\cdot)] \triangleright W & \overset{\tau}{\longmapsto} & \Delta[c \mapsto (k, \mathsf{err})] \triangleright W_0 \text{ if } k > 0 \\
\Delta[c \mapsto (k,\cdot)] \triangleright V & \overset{\tau}{\longmapsto} & \Delta[c \mapsto (2, \mathsf{err})] \triangleright V_1 \text{ if } 0 < k \leq 2 \\
\Delta[c \mapsto (k,\cdot)] \triangleright V & \overset{\tau}{\longmapsto} & \Delta[c \mapsto (k, \mathsf{err})] \triangleright V_1 \text{ if } k > 2 \\
 & & \\
\Delta[c \mapsto (k,\cdot)] \triangleright W & \overset{\tau}{\longmapsto} & \Delta[c \mapsto (3, \mathsf{err})] \triangleright W_{ok} \text{ if } 0 < k \leq 3 \\
\Delta[c \mapsto (k,\cdot)] \triangleright W & \overset{\tau}{\longmapsto} & \Delta[c \mapsto (k, \mathsf{err})] \triangleright W_{ok} \text{ if } k > 3 \\
\Delta[c \mapsto (k,\cdot)] \triangleright V & \overset{\tau}{\longmapsto} & \Delta[c \mapsto (3, \mathsf{err})] \triangleright V_{ok} \text{ if } 0 < k \leq 3 \\
\Delta[c \mapsto (k,\cdot)] \triangleright V & \overset{\tau}{\longmapsto} & \Delta[c \mapsto (k, \mathsf{err})] \triangleright V_{ok} \text{ if } k > 3 \\
 & & \\
\Delta \triangleright W & \overset{d?v}{\longmapsto} & \mathrm{upd}_{d?v}(\Delta) \triangleright W \\
\Delta \triangleright V & \overset{d?v}{\longmapsto} & \mathrm{upd}_{d?v}(\Delta) \triangleright V
\end{array}$$

$$\begin{array}{lcl}
\Delta[(c \mapsto (1, v_0)] \triangleright W_0 & \stackrel{\tau}{\longmapsto} & \Delta[(c \mapsto (3, \mathsf{err})] \triangleright W_{\mathsf{err}} \\
\Delta[c \mapsto (2, v_1)] \triangleright V_1 & \stackrel{\tau}{\longmapsto} & \Delta[(c \mapsto (3, \mathsf{err})] \triangleright V_{\mathsf{err}}
\end{array}$$

$$\begin{array}{lcl}
\Delta[c \mapsto (k, \cdot)] \triangleright W_0 & \stackrel{\tau}{\longmapsto} & \Delta[c \mapsto (3, \mathsf{err})] \triangleright W_{\mathsf{err}} \text{ if } 0 < 3 \leq k \\
\Delta[c \mapsto (k, \cdot)] \triangleright V_1 & \stackrel{\tau}{\longmapsto} & \Delta[c \mapsto (3, \mathsf{err})] \triangleright V_{\mathsf{err}} \text{ if } 0 < 3 \leq k
\end{array}$$

$$\begin{array}{lcl}
\Delta[c \mapsto (k, \cdot)] \triangleright W_0 & \stackrel{\tau}{\longmapsto} & \Delta[c \mapsto (k, \mathsf{err})] \triangleright W_{\mathsf{err}} \text{ if } k > 3 \\
\Delta[c \mapsto (k, \cdot)] \triangleright V_1 & \stackrel{\tau}{\longmapsto} & \Delta[c \mapsto (k, \mathsf{err})] \triangleright V_{\mathsf{err}} \text{ if } k > 3
\end{array}$$

$$\begin{array}{lcl}
\Delta \triangleright W_0 & \stackrel{d?w}{\longmapsto} & \mathrm{upd}_{d?w}(\Delta) \triangleright W_0 \\
\Delta \triangleright V_1 & \stackrel{d?v}{\longmapsto} & \mathrm{upd}_{d?w}(\Delta) \triangleright V_1
\end{array}$$

$$\begin{array}{lcl}
\Delta[c \mapsto (k, \cdot)] \triangleright W_{ok} & \stackrel{\tau}{\longmapsto} & \Delta[c \mapsto k, \cdot] \triangleright W_{\mathsf{err}} \text{ if } k > 3 \\
\Delta[c \mapsto (k, \cdot)] \triangleright V_{ok} & \stackrel{\tau}{\longmapsto} & \Delta[c \mapsto k, \cdot] \triangleright V_{\mathsf{err}} \text{ if } k > 3
\end{array}$$

$$\begin{array}{lcl}
\Delta \triangleright W_{ok} & \stackrel{d?w}{\longmapsto} & \mathrm{upd}_{d?w}(\Delta) \triangleright W_{ok} \\
\Delta \triangleright V_{ok} & \stackrel{d?w}{\longmapsto} & \mathrm{upd}_{d?w}(\Delta) \triangleright V_{ok}
\end{array}$$

□

4. Full abstraction

In this section, we show that the co-inductive proof method based on the bisimulation of the previous section is sound with respect to the contextual equivalence of Section 2.4; this is the subject of Section 4.1. Moreover it is complete for a large class of systems. This class is isolated in Section 4.2.1, and the completeness result is then given in Section 4.2.2.

4.1. **Soundness.** In this section we prove that (weak) bisimulation equivalence is contained in reduction barbed congruence. The main difficulty is in proving the contextuality of the bisimulation equivalence. But first some auxiliary results.

Lemma 4.1. [Update of Channel Environments] If $\Gamma \triangleright W \Longrightarrow \Gamma' \triangleright W'$ then $\Gamma \leq \Gamma'$.

Proof. See the Appendix, Page 51.

□

Below we report a result on channel exposure for bisimilarity; a similar result for reduction barbed congruence will also be proved, in Proposition 4.13.

Lemma 4.2. [Channel exposure w.r.t. $\approx$] Whenever $\Gamma_1 \triangleright W_1 \approx \Gamma_2 \triangleright W_2$ then $\Gamma_1 \vdash c$: **idle** if and only if $\Gamma_2 \vdash c$: **idle**.

Proof. See the Appendix, Page 52 □

In order to prove that weak bisimulation is sound with respect to reduction barbed congruence we need to show that $\approx$ is preserved by parallel composition.

Theorem 4.3. [$\approx$ is contextual] Suppose $\Gamma_1 \triangleright W_1 \approx \Gamma_2 \triangleright W_2$. Then for any system term W, $\Gamma_1 \triangleright (W_1 \mid W) \approx \Gamma_2 \triangleright (W_2 \mid W)$.

Proof. Let the relation $\mathcal{S}$ over configurations be defined as follows:

$$\{(\Gamma_1 \triangleright W_1 \mid W\,,\ \Gamma_2 \triangleright W_2 \mid W) :\ \ \Gamma_1 \triangleright W_1 \approx \Gamma_2 \triangleright W_2\,\}$$

It is sufficient to show that $\mathcal{S}$ is a bisimulation in the extensional semantics. To do so, by symmetry, we need to show that an arbitrary extensional action

$$\Gamma_1 \triangleright W_1 \mid W \stackrel{\alpha}{\longmapsto} \widehat{\Gamma_1} \triangleright \widehat{W_1} \tag{4.1}$$

can be matched by $\Gamma_2 \triangleright W_2 \mid W$ via a corresponding weak extensional action.

The action (4.1) can be inferred by any of the six rules in Table 6. We consider only one case, the most interesting one (Shh). So here α is τ and $\Gamma_1 \triangleright W_1 \mid W \xrightarrow{c!v} \widehat{W_1}$, for some c and v, and $\widehat{\Gamma_1} = \mathrm{upd}_{c!v}(\Gamma)$. This transition in turn can always be inferred by an application of the rule (Sync), or its symmetric counterpart, from Table 2. Here we only consider the former case; the proof for the second case is slightly different, though it uses the same proof strategies illustrated below. For the case we are considering, we have that

- $\Gamma_1 \triangleright W_1 \xrightarrow{c!v} W_1'$
- $\Gamma_1 \triangleright W \xrightarrow{c?v} W'$
- $\widehat{W_1} = W_1' \mid W'$

By an application of rule (Shh) it follows that $\Gamma_1 \triangleright W_1 \stackrel{\tau}{\longmapsto} \widehat{\Gamma_1} \triangleright W_1'$. Since $\Gamma_1 \triangleright W_1 \approx \Gamma_2 \triangleright W_2$, there is $\Gamma_2' \triangleright W_2'$ such that $\Gamma_2 \triangleright W_2 \Longmapsto \Gamma_2' \triangleright W_2'$ and $\Gamma_1' \triangleright W_1' \approx \Gamma_2' \triangleright W_2'$. Note that Lemma 4.2 ensures that whenever $\Gamma_1 \vdash d : \mathbf{exp}$ then also $\Gamma_2 \vdash d : \mathbf{exp}$, for any channel d . Similarly, if $\widehat{\Gamma_1} \vdash d : \mathbf{exp}$ then $\widehat{\Gamma_2} \vdash d : \mathbf{exp}$. That is, Γ_1 agrees with Γ_2 on the exposure state of each channel; the same applies to $\widehat{\Gamma_1}$ and $\widehat{\Gamma_2}$.

Further, recall that $\widehat{\Gamma_1} = \mathrm{upd}_{c!v}(\Gamma_1)$. Therefore we have that, for any channel $d \neq c$, $\Gamma_1 \vdash d : \mathbf{exp}$ iff $\widehat{\Gamma_1} \vdash d : \mathbf{exp}$; for channel c, we have that $\widehat{\Gamma_1} \vdash c : \mathbf{exp}$. That is, the exposure states of Γ_1 and $\widehat{\Gamma_1}$ differ only in the entry at channel c, and only if such a channel was idle in Γ_1.

Since Γ_1 and $\widehat{\Gamma_1}$ agree with $\Gamma_2, \widehat{\Gamma_2}$, respectively, on the exposure state of each channel, it has also to be that the exposure states of Γ_2 and $\widehat{\Gamma_2}$ differ only at the entry at channel c, and only when the latter is idle in Γ_2; formally $\Gamma_2 \vdash d : \mathbf{exp}$ iff $\widehat{\Gamma_2} \vdash d : \mathbf{exp}$ when $d \neq c$, and $\widehat{\Gamma_2} \vdash c : \mathbf{exp}$.

Next we show that the action $\Gamma_1 \triangleright W_1 \mid W \stackrel{\tau}{\longmapsto} \widehat{\Gamma_1} \triangleright W_1' \mid W'$ can be matched by a weak action $\Gamma_2 \triangleright W_2 \mid W \Longmapsto \widehat{\Gamma_2} \triangleright W_2' \mid W'$. Since $\widehat{\Gamma_1} \triangleright W_1' \approx \widehat{\Gamma_2} \triangleright W_2'$, the above statement would imply that $(\widehat{\Gamma_1} \triangleright W_1' \mid W)\,\mathcal{S}\,(\widehat{\Gamma_2} \triangleright W_2' \mid W)$, which is exactly what we want to prove. There are two possible cases, according to whether $\Gamma_1 \triangleright W$ is able to detect a value broadcast along channel c:

(1) $\neg\mathsf{rcv}(\Gamma_1 \triangleright W, c)$. By Lemma 2.9(1), in the transition $\Gamma_1 \triangleright W \xrightarrow{c?v} W'$ it must be that $W' = W$. We have to show that the transition $\Gamma_2 \triangleright W_2 \Longmapsto \widehat{\Gamma_2} \triangleright W_2'$ implies that $\Gamma_2 \triangleright W_2 \mid W \Longmapsto \widehat{\Gamma_2} \triangleright W_2' \mid W$. To this end, we prove a stronger statement: whenever we have a sequence of transitions

$$\Gamma^0 \triangleright V^0 \stackrel{\tau}{\longmapsto} \Gamma^1 \triangleright V^1 \stackrel{\tau}{\longmapsto} \cdots \stackrel{\tau}{\longmapsto} \Gamma^n \triangleright V^n$$

of arbitrary length $n \geq 0$, and such that for any $d \neq c$, $\Gamma^0 \vdash d : \mathbf{exp}$ if and only if $\Gamma^n \vdash d : \mathbf{exp}$, and $\neg\mathsf{rcv}(\Gamma_0 \triangleright W)$. Then

$$\Gamma^0 \triangleright V^0 \mid W \stackrel{\tau}{\longmapsto} \Gamma^1 \triangleright V^1 \mid W \stackrel{\tau}{\longmapsto} \cdots \stackrel{\tau}{\longmapsto} \Gamma^n \triangleright V^n \mid W$$

Further, $\neg\mathsf{rcv}(\Gamma^n \triangleright W, c)$. By choosing $\Gamma^0 \triangleright V^0 = \Gamma_2 \triangleright W_2$ and $\Gamma^n \triangleright V^n = \widehat{\Gamma_2} \triangleright W_2'$ we obtain that $\Gamma_2 \triangleright W_2 \mid W \Longmapsto \widehat{\Gamma_2} \triangleright W_2' \mid W$.

The proof of the aforementioned statement is by induction on n.

(a) If $n = 0$ then there is nothing to prove.

(b) Let $n > 0$. By inductive hypothesis we assume that the statement is true for $n-1$. By Lemma 4.1 we know that $\Gamma^0 \leq \Gamma^{n-1} \leq \Gamma^n$. Let $d \neq c$; if $\Gamma^0 \vdash d : \mathbf{exp}$, then $\Gamma^{n-1} \vdash d : \mathbf{exp}$ since $\Gamma^0 \leq \Gamma^{n-1}$. Conversely, if $\Gamma^{n-1} \vdash d : \mathbf{exp}$ then $\Gamma^{n-1} \leq \Gamma^n$ implies that $\Gamma^n \vdash d : \mathbf{exp}$, and by hypothesis we get that $\Gamma^0 \vdash d : \mathbf{exp}$.
Therefore we can apply the inductive hypothesis to obtain the sequence of transitions

$$\Gamma^0 \triangleright V^0 \mid W \stackrel{\tau}{\longmapsto} \Gamma^1 \triangleright V^1 \mid W \stackrel{\tau}{\longmapsto} \cdots \stackrel{\tau}{\longmapsto} \Gamma^{n-1} \triangleright V^{n-1} \mid W$$

and infer that $\neg\mathsf{rcv}(\Gamma^{n-1} \triangleright W, c)$. Consider now the transition $\Gamma^{n-1} \triangleright V^{n-1} \stackrel{\tau}{\longmapsto} \Gamma^n \triangleright V^n$. There are different ways in which this extensional transition could have been inferred:

- if this transition has been obtained by an application of Rule (TauExt) of Table 6, then we have that $\Gamma^{n-1} \triangleright V^{n-1} \xrightarrow{\tau} V^n$, and $\Gamma^n = \mathrm{upd}_\tau(\Gamma^{n-1})$. By Rule (TauPar) we also have that $\Gamma^{n-1} \triangleright V^{n-1} \mid W \xrightarrow{\tau} V^n \mid W$, which can now be translated in an extensional τ-action $\Gamma^{n-1} \triangleright V^{n-1} \mid W \stackrel{\tau}{\longmapsto} \Gamma^n \triangleright V^n \mid W$ via an application of Rule (TauExt).
- if the transition has been obtained by an application of Rule (Shh) of Table 6, then $\Gamma^{n-1} \triangleright V^{n-1} \xrightarrow{d!w} V^n$, and $\Gamma^n = \mathrm{upd}_{d!w}(\Gamma^{n-1})$. Let us perform a case analysis on the channel d:
  - If $d = c$, then since $\neg\mathsf{rcv}(\Gamma^{n-1} \triangleright W, c)$, Lemma 2.9(1) ensures that we have the transition $\Gamma^{n-1} \triangleright W \xrightarrow{c?w} W$. Note also that now $\Gamma^n \vdash c : \mathbf{exp}$, so that it follows $\neg\mathsf{rcv}(\Gamma^n \triangleright W, c)$. Now by applying Rule (Sync) to the transitions $\Gamma^{n-1} \triangleright V^{n-1} \xrightarrow{c!w} V^n$ and $\Gamma^{n-1} \triangleright W \xrightarrow{c?w} W$ we obtain $\Gamma^{n-1} \triangleright V^{n-1} \mid W \xrightarrow{d!v} V^n \mid W$. The latter can be converted into an extensional τ-transition $\Gamma^{n-1} \triangleright V^{n-1} \mid W \stackrel{\tau}{\longmapsto} \Gamma^n \triangleright V^n \mid W$ using Rule (Shh) and the fact that $\Gamma^n = \mathrm{upd}_{c!w}(\Gamma^{n-1})$.
  - It remains to check the case $d \neq c$. First note that if we have $\Gamma^{n-1} \vdash c : \mathbf{exp}$ then also $\Gamma^n \vdash c : \mathbf{exp}$ (since $\Gamma^{n-1} \leq \Gamma^n$), so that $\neg\mathsf{rcv}(\Gamma^n \triangleright W, c)$. On the other hand, if $\Gamma^{n-1} \vdash c : \mathbf{idle}$, we can still prove that $\neg\mathsf{rcv}(\Gamma^n \triangleright W, c)$ via an induction on the structure of W.[4]
    Finally, note that since $\Gamma^n = \mathrm{upd}_{d!w}(\Gamma^{n-1})$ implies that $\Gamma^n \vdash d : \mathbf{exp}$. By hypothesis we get that $\Gamma^0 \vdash d : \mathbf{exp}$, which leads to $\Gamma^{n-1} \vdash d : \mathbf{exp}$ (recalling that $\Gamma^0 \leq \Gamma^{n-1}$). Therefore we have that $\neg\mathsf{rcv}(\Gamma^{n-1} \triangleright W, d)$, and by Lemma 2.9(1) we obtain that $\Gamma^{n-1} \triangleright W \xrightarrow{d?v} W$. Now we can proceed as in the case $d = c$ to infer the extensional transition $\Gamma^{n-1} \triangleright V^{n-1} \mid W \stackrel{\tau}{\longmapsto} \Gamma^n \triangleright V^n \mid W$.

(2) Suppose now that $\mathsf{rcv}(\Gamma_1 \triangleright W, c)$. By Lemma 2.9(2) the transition $\Gamma_1 \triangleright W \xrightarrow{c?v} W'$ leads to $W' \neq W$. Also, in this case we have that $\Gamma_1 \vdash c : \mathbf{idle}$, which also gives $\Gamma_2 \vdash c : \mathbf{idle}$ by

[4] Intuitively, we just need to check that there are no unguarded receivers along channel c appearing in W.

Lemma 4.2. Since we have $\widehat{\Gamma_2} \vdash c : \mathbf{exp}$, it has to be the case that we can unfold the weak transition $\Gamma_2 \triangleright W_2 \Longrightarrow \widehat{\Gamma_2} \triangleright W_2'$ as

$$\Gamma_2 \triangleright W_2 \Longrightarrow \Gamma_2^{\mathrm{pre}} \triangleright W_2^{\mathrm{pre}} \stackrel{\tau}{\longmapsto} \Gamma_2^{\mathrm{post}} \triangleright W_2^{\mathrm{post}} \Longrightarrow \widehat{\Gamma_2} \triangleright W_2'$$

where $\Gamma_2^{\mathrm{pre}} \vdash c : \mathbf{idle}$ and $\Gamma_2^{\mathrm{post}} \vdash c : \mathbf{exp}$. Note also that Lemma 4.1 ensures that, for any channel $d \neq c$, $\Gamma_2 \vdash d : \mathbf{exp}$ implies $\Gamma_2^{\mathrm{pre}} \vdash d : \mathbf{exp}$, and $\Gamma_2^{\mathrm{pre}} \vdash d : \mathbf{exp}$ implies $\widehat{\Gamma_2} \vdash d : \mathbf{exp}$, which by hypothesis leads to $\Gamma_2 \vdash d : \mathbf{exp}$. Similarly we can show that $\Gamma_2^{\mathrm{post}} \vdash d : \mathbf{exp}$ if and only if $\widehat{\Gamma_2} \vdash d : \mathbf{exp}$. That is, Γ_2, Γ_2' agree with $\Gamma_2^{\mathrm{pre}}, \Gamma_2^{\mathrm{post}}$ on the exposure state of each channel, respectively. Now, in a way similar to the first case, we can prove that we have the following transitions:

- $\Gamma_2 \triangleright W_2 \mid W \Longrightarrow \Gamma_2^{\mathrm{pre}} \triangleright W_2^{\mathrm{pre}} \mid W$,
- $\Gamma_2^{\mathrm{post}} \triangleright W_2^{\mathrm{post}} \mid W' \Longrightarrow \widehat{\Gamma_2} \triangleright W_2' \mid W'$.

so that it remains to show that $\Gamma_2^{\mathrm{pre}} \triangleright W_2^{\mathrm{pre}} \mid W \stackrel{\tau}{\longmapsto} \Gamma_2^{\mathrm{post}} \triangleright W_2^{\mathrm{post}} \mid W'$. Note that, since $\Gamma_2^{\mathrm{pre}} \vdash c : \mathbf{idle}$ and $\Gamma_2^{\mathrm{post}} \vdash c : \mathbf{exp}$, it has to be the case that the transition $\Gamma_2^{\mathrm{pre}} \triangleright W_2^{\mathrm{pre}} \stackrel{\tau}{\longmapsto} \Gamma_2^{\mathrm{post}} \triangleright W_2^{\mathrm{post}}$ has been induced by the intensional one $\Gamma_2^{\mathrm{pre}} \triangleright W_2^{\mathrm{pre}} \xrightarrow{c!w} W_2^{\mathrm{post}}$, and $\Gamma_2^{\mathrm{post}} = \mathrm{upd}_{c!w}(\Gamma_2^{\mathrm{pre}})$.

Now note that, since $\Gamma_1 \triangleright W \xrightarrow{c?v} W'$ we also have that $\Gamma_1 \triangleright W \xrightarrow{c!w} W'$ by Lemma 2.9(2). Finally, note that for any channel c, $\Gamma_1 \vdash d : \mathbf{exp}$ iff $\Gamma_2 \vdash d : \mathbf{exp}$ (as $\Gamma_1 \triangleright W_1 \approx \Gamma_2 \triangleright W_2$) iff $\Gamma_2^{\mathrm{pre}} \vdash d : \mathbf{exp}$. By Proposition 2.12 it follows that $\Gamma_1 \triangleright W \xrightarrow{c!w} W'$ implies $\Gamma_2^{\mathrm{pre}} \triangleright W \xrightarrow{c?w} W'$. We can now apply Rule (Sync) to such a transition, and the transition $\Gamma_2^{\mathrm{pre}} \triangleright W_2^{\mathrm{pre}} \xrightarrow{c!w} W_2^{\mathrm{post}}$, to infer $\Gamma_2^{\mathrm{pre}} \triangleright W_2^{\mathrm{pre}} \mid W \xrightarrow{c!w} W_2^{\mathrm{post}} \mid W'$. The last transitions induces the extensional action $\Gamma_2^{\mathrm{post}} \triangleright W_2^{\mathrm{post}} \mid W' \stackrel{\tau}{\longmapsto} \Gamma_2^{\mathrm{post}} \triangleright W_2^{\mathrm{post}} \mid W'$, as we wanted to prove.

We have built the sequence of transitions

$$\Gamma_2 \triangleright W_2 \mid W \Longrightarrow \Gamma_2^{\mathrm{pre}} \triangleright W_2^{\mathrm{pre}} \mid W \stackrel{\tau}{\longmapsto} \Gamma_2^{\mathrm{post}} \triangleright W_2^{\mathrm{post}} \mid W' \Longrightarrow \widehat{\Gamma_2} \triangleright W_2' \mid W'$$

which can be synthesised as $\Gamma_2 \triangleright W_2 \mid W \Longrightarrow \widehat{\Gamma_2} \triangleright W_2' \mid W'$, which is exactly the transition that we wanted to derive.

□

Theorem 4.4. [Soundness] $C_1 \approx C_2$ implies $C_1 \simeq C_2$.

Proof. It suffices to prove that bisimilarity is reduction-closed, barb preserving and contextual.

Reduction Closure: Note that if $C_1 \rightarrow\!\triangleright C_1'$, then we have two possible cases; either $C_1 \rightarrow\!\triangleright_i C_1'$ or $C_1 \rightarrow\!\triangleright_\sigma C_1'$. If $C_1 \rightarrow\!\triangleright_i C_1'$ then it is not difficult to see that $C_1 \stackrel{\tau}{\longmapsto} C_1'$ (see Remark 3.1). Similarly, if $C_1 \rightarrow\!\triangleright_\sigma C_1'$ then $C_1 \stackrel{\sigma}{\longmapsto} C_1'$. Since $C_1 \approx C_2$, it follows that there exists C_2' such that $C_2 \Longrightarrow C_2'$ (respectively, $C_2 \stackrel{\sigma}{\Longrightarrow} C_2'$) with $C_1' \approx C_2'$. By Remark 3.1 the last transition can be rewritten as a sequence of reductions $C_2 \rightarrow\!\triangleright_i^* C_2'$ (respectively, $C_2 \rightarrow\!\triangleright_i^* \rightarrow\!\triangleright_\sigma \rightarrow\!\triangleright_i^* C_2'$), from which it follows $C_2 \rightarrow\!\triangleright^* C_2'$,

Barb Preservation: Let $C_1 = \Gamma_1 \triangleright W_1$ and $C_2 = \Gamma_2 \triangleright W_2$. Suppose that $C_1 \downarrow c$ for some channel c; by definition we have that $\Gamma - 1 \vdash c : \mathbf{exp}$. By Lemma 4.2 we also have that $\Gamma_2 \vdash c : \mathbf{exp}$. This ensures that $C_2 \downarrow_c$, and more generally $C_2 \Downarrow_c$.

Contextuality: contextuality has already been proved as Theorem 4.3.

□

4.2. **Completeness.** Having proved soundness, it remains to check whether our bisimulation proof technique is also complete with respect to reduction barbed congruence; that is, whenever we have $\Gamma_1 \triangleright W_1 \simeq \Gamma_2 \triangleright W_2$, then there exists a bisimulation that contains the pair $(\Gamma_1 \triangleright W_1, \Gamma_2 \triangleright W_2)$. Unfortunately, this is not true for arbitrary configurations, as shown by the following Example:

Example 4.5. Let $\Gamma_1 \vdash c : \mathbf{exp}$, Γ_1 and $\Gamma_2 \vdash c : \mathbf{idle}$ and consider the two configurations $C_1 = \Gamma_1 \triangleright \nu d{:}(0,\cdot).(d[x].\mathsf{nil})$ and $C_2 = \Gamma_2 \triangleright c!\langle v\rangle \mid \nu d{:}(0,\cdot).(d[x].\mathsf{nil})$. Note that both configurations include an active receiver placed along an idle, restricted channel. The presence of such an active receiver is somewhat problematic, as it does not allow the passage of time in both configurations, according to our definition of timed reductions. Indeed, the reader can check that, in the intensional semantics, no transition $\xrightarrow{\sigma}$ is defined for a configuration of the form $\Gamma[d \mapsto (0,\cdot)] \triangleright d[x].P$; consequently, σ-transitions are not allowed for the configuration $\Gamma \triangleright \nu d : (0, v).(d[x].P$ either. Similarly, weak σ-transitions are note enabled in C_2.

Now note that, since any occurrence of channel d is restricted in both C_1, C_2, we cannot enable the passage of time for them via the composition with a system term T. That is, for any system term T, and configuration $\widehat{C_1}, \widehat{C_2}$, such that $C_1 \mid T \rightarrow\!\triangleright_i^* \widehat{C_1}$, $C_2 \mid T \rightarrow\!\triangleright_i^* \widehat{C_2}$, we have that $\widehat{C_1} \not\rightarrow\!\triangleright_\sigma$ and $\widehat{C_2} \not\rightarrow\!\triangleright\, \sigma$.

Now it is not difficult to show that $C_1 \simeq C_2$. At least informally, the only difference between these two configurations lies in the exposure state of channel c, and in the fact that C_2 can broadcast along channel c. Such a broadcast ensures that the strong barb at channel c, enabled in C_1, can be matched by a weak barb enabled at C_2. On the other hand, the difference in the exposure state of channel c in C_1, C_2 could be detected via a test T which contains an exposure check $\exp(c)$; however, this construct requires the passage of time in order to determine that channel c is free (exposed) in $C_1 \mid T$ (respectively, $C_2 \mid T$). But, as we have already noticed, time is not allowed to pass in such configurations. Formally, to prove $C_1 \simeq C_2$ it suffices to show that the relation

$$\{ \quad (\Delta \triangleright \nu d{:}(0,\cdot).(d[x].\mathsf{nil}), \Delta' \triangleright \nu d{:}(0,\cdot).(c!\langle v\rangle \mid d[x].P) \quad \mid$$
$$\mid \quad \Delta \vdash c : \mathbf{exp}, \Delta \vdash d : \mathbf{exp} \text{ iff } \Delta' \vdash d : \mathbf{exp} \text{ for } d \neq c \quad \}$$

is barb-preserving, reduction closed and contextual.

Therefore we have shown that $C_1 \simeq C_2$; however, $\Gamma_1 \vdash c : \mathbf{idle}$, while $\Gamma_2 \vdash c : \mathbf{exp}$. Therefore, by Lemma 4.2 it also has to be $C_1 \not\approx C_2$. □

4.2.1. *Well-formed systems.* The counterexample to completeness illustrated in Example 4.5 relies on the existence of configurations which do not let time pass. These can be built by placing an active receiver along an idle, restricted channel. However, such configurations are not interesting per se, as it is counter-intuitive to allow wireless stations to receive a value along a channel, when there is no value being transmitted.

It is interesting, in fact, to ask ourselves if our proof methodology based on bisimulations is complete, if we were to restrict our focus to a setting where active receivers along idle channels were explicitly forbidden. These take the name of *well-formed* configurations, and can be defined as below:

Definition 4.6. [Well-formedness] The set of well-formed configurations WNets is the least set such that

$$\begin{array}{rcl} \Gamma \triangleright P \in \text{WNets} & & \text{for all processes } P \\ \Gamma \vdash c : \mathbf{exp} & \text{implies} & \Gamma \triangleright c[x].P \in \text{WNets} \\ \Gamma \triangleright W_1, \Gamma \triangleright W_2 \in \text{WNets} & \text{implies} & \Gamma \triangleright W_1 \mid W_2 \in \text{WNets} \\ \Gamma[c \mapsto (n, v)] \triangleright W \in \text{WNets} & \text{implies} & \Gamma \triangleright \nu c : (n, v).W \in \text{WNets} \end{array}$$

□

A configuration $\Gamma \triangleright W$ is well-formed if it does not contain any receiving station along an idle channel. Note that the configurations from Example 4.5 are not well-formed. Clearly, well-formed configurations are preserved at runtime.

Lemma 4.7. Suppose C is well-formed and $C \rightarrow\!\!\triangleright C'$. Then C' is also well-formed.

Proof. See the Appendix, Page 52. □

The main property of well-formed systems is that they allow the passage of time, so long as all internal activity has ceased:

Proposition 4.8. [Patience] Let C be a well-formed configuration for which there is no C' such that $C \rightarrow\!\!\triangleright_i C'$; then $C \rightarrow\!\!\triangleright_\sigma C''$, for some configuration C''.

Proof. Details for the most important cases are given in the Appendix; see Page 52. □

However, Patience alone does not preclude the possibility of exhibiting a configuration in which time never passes. In fact, it only ensures the passage of time when instantaneous reduction are not possible anymore. However, it could be the case that a configuration C enables an infinite sequence of instantaneous reductions, and by maximal progress (Proposition 2.11) the passage of time would be forbidden. As we will prove presently, this phenomenon does not arise for CCCP configurations; we recall in fact that, in recursive processes of the form $\mathsf{fix}\, X.P$, we require all free occurrences of the process variable X in P to be guarded by a time-consuming construct. This limitation is sufficient to prevent the existence of configurations which do not allow time to pass; further, it is also necessary, as shown by the following example.

Example 4.9. Suppose we remove the constraint in the syntax that process variables have to be guarded by time-consuming constructs in fixed point processes. Let W denote the code $\mathsf{fix}\, X.(\tau.X)$. Then we have an infinite sequence of internal actions

$$\Gamma \triangleright W \rightarrow\!\!\triangleright_i C_1 \rightarrow\!\!\triangleright_i \ldots C_k \rightarrow\!\!\triangleright_i$$

Indeed one can show that if $\Gamma \triangleright W \rightarrow\!\!\triangleright^* C'$ then $C' \rightarrow\!\!\triangleright_i$. Maximal progress then ensures that $C' \not\rightarrow\!\!\triangleright_\sigma$.

□

Example 4.10. Again, suppose we remove the constraint on guarded recursion in the syntax of CCCP. Then our bisimulation proof principle would not be complete; to see this, it is sufficient to consider the two configurations $\Gamma \triangleright \mathsf{fix}\, X.(\tau.X)$ and $\Gamma' \triangleright \mathsf{fix}\, X.(\tau.X) \mid c!\langle v \rangle$, where $\Gamma \vdash c : \mathbf{exp}$ and $\Gamma' \vdash c : \mathbf{idle}$. By Lemma 4.2 these two configurations are not bisimilar, as they differ in the exposure state of channel c. On the other hand, none of these two configurations allow the passage of time. As we have already argued in Example 4.5, when the passage of time is not allowed in a configuration, it is not possible to provide a context that determines the exposure state of a channel. Then it is not

difficult to show that $\Gamma \triangleright \mathsf{fix}\, X.(\tau.X) \simeq \Gamma' \triangleright \mathsf{fix}\, X.(\tau.X) \mid c!\langle v\rangle$. This can be done by simply showing that the relation

$$\begin{aligned}\mathcal{S} \;=\; \{&(\Delta \triangleright \mathsf{fix}\, X.\tau.X, \Delta' \triangleright \mathsf{fix}\, X.\tau.X \mid c!\langle v\rangle), (\Delta_c \triangleright \mathsf{fix}\, X.\tau.X, \Delta'_c \triangleright \mathsf{fix}\, X.\tau.X \mid \sigma^{\delta_v}) \mid \\ &\Delta \vdash d : \mathbf{exp} \text{ if and only if } \Delta' \vdash d : \mathbf{exp}, d \neq c, \\ &\Delta_c \vdash d : \mathbf{exp} \text{ if and only if } \Delta'_c \vdash d : \mathbf{exp}, \text{with } d \text{ arbitrary } \}\end{aligned}$$

is a bisimulation. □

Let us state precisely what we mean when we say that infinite sequences of instantaneous reductions are not allowed in our calculus. In practice, we give a slightly stronger definition, requiring that the amount of instantaneous reductions that can be performed in sequence by a configuration C is bounded.

Definition 4.11. [Well-timed configurations] A configuration C is *well-timed*, [32], if there exists an upper bound $k \in \mathbb{N}$ such that whenever $C\ (\twoheadrightarrow_i)^h\ C'$ for some $h \geq 0$, then $h \leq k$. □

Contrarily to well-formedness, which is a simple syntactic constraint, *well-timedness* means that the designer of the network has to ensure that the code placed at the station nodes can never lead to divergent behaviour. As we already argued, however, the constraint we have placed on the syntax of system terms that each recursive definition is weakly guarded in P, is sufficient to ensure well-timedness. One simple method for ensuring this is to only use recursive definitions $\mathsf{fix}\, X.P$ where X is weakly guarded in P; that is, every occurrence of X is within an input, output or time delay prefix, or it is included within a branch of a matching construct. These are exactly the conditions that we placed for recursion variables when defining our calculus. Thus, we would expect every configuration in our calculus to be well-timed.

Proposition 4.12. Any configuration $\Gamma \triangleright W$ is well-timed.

Proof. See the Appendix, Page 53. □

Next we prove a very useful result for well-defined configurations; the proof emphasises the roles of well-formedness and well-timedness in the configurations being tested.

Proposition 4.13. Suppose $\Gamma_1 \triangleright W_1 \simeq \Gamma_2 \triangleright W_2$, where both are well-formed. Then $\Gamma_1 \vdash c : \mathbf{idle}$ implies $\Gamma_2 \vdash c : \mathbf{idle}$.

Proof. Let $\Gamma_1 \triangleright W_1 \simeq \Gamma_2 \triangleright W_2$ and suppose $\Gamma_1 \vdash c : \mathbf{idle}$ for some channel c. Consider the testing code:

$$T = [\exp(c)]\mathsf{nil}, \mathit{eureka}!\langle \mathit{ok}\rangle$$

From the definition of $\simeq$ we know that $\Gamma_1 \triangleright W_1 \mid T \simeq \Gamma_2 \triangleright W_2 \mid T$. Since $\Gamma_1 \triangleright W_1$ is well-timed, by definition there is a configuration C such that $\Gamma_1 \triangleright W_1 \twoheadrightarrow_i^* C$ and $C \not\twoheadrightarrow_i$. Because $\Gamma_1 \triangleright W_1$ is well-formed so is C. By Proposition 4.8 there is a configuration C' such that $C \twoheadrightarrow_\sigma C'$. Let $C' = \Gamma' \triangleright W'$, for some Γ' and W'. Now, if we define $C'' = \mathrm{upd}_{\mathit{eureka}!\mathit{ok}}(\Gamma') \triangleright W'$ and $T' = \sigma.\mathit{eureka}!\langle \mathit{ok}\rangle$, it is easy to see that there exists a sequence of reductions of the following shape:

$$\Gamma_1 \triangleright W_1 \mid T \twoheadrightarrow_i \Gamma_1 \triangleright W_1 \mid T' \twoheadrightarrow_i^* C \mid T' \twoheadrightarrow_\sigma C' \mid \mathit{eureka}!\langle \mathit{ok}\rangle \twoheadrightarrow_i C'' \mid \sigma^{\delta_{\mathrm{ok}}}$$

where $C'' \mid \sigma^{\delta_{\mathrm{ok}}} \downarrow_{\mathit{eureka}}$. By definition this implies that $\Gamma_1 \triangleright W_1 \mid T \Downarrow_{\mathit{eureka}}$.

Note that the existence of the sequence of reductions above relies on the fact that $\Gamma_1 \triangleright W_1$ is well-timed.The timed transition $C \mid T' \twoheadrightarrow_\sigma C' \mid \mathit{eureka}!\langle \mathit{ok}\rangle$ in such a sequence is derived from the timed transitions performed by their components; if C were not able to perform a σ-transition, in fact, we would have not been able to derive the timed reduction for the overall configuration $C \mid T'$.

Since $\Gamma_1 \triangleright W_1 \mid T \simeq \Gamma_2 \triangleright W_2 \mid T$ we also have that $\Gamma_2 \triangleright W_2 \mid T \Downarrow_{eureka}$. This is only possible if

$$\Gamma_2 \triangleright W_2 \mid T \;\; \rightarrow\!\triangleright_i^* \;\; \Gamma_2' \triangleright W_2' \mid T' \;\; \rightarrow\!\triangleright_i^* \rightarrow\!\triangleright_\sigma \rightarrow\!\triangleright_i^* \;\; \Gamma_2'' \triangleright W_2'' \mid \sigma^{\delta_{ok}}$$

where Γ_2' is a channel environment such that $\Gamma_2' \vdash c : \mathbf{idle}$. From Lemma 4.1 (recall that τ-extensional actions coincide with instantaneous reductions) we get the required $\Gamma_2 \vdash c : \mathbf{idle}$. □

We remark once again that restricting our attention to well-formed configurations is crucial in order to ensure the validity of Proposition 4.13. In fact, in Example 4.5 we have already provided an example of two (ill-formed) configurations which are reduction barbed congruent, but which differ in the exposure state of a channel.

Another important property that we will need from well-formed configurations concerns the definition of reduction barbed congruence itself; the reduction closure property which we used to define $\simeq$ can be strengthened by requiring instantaneous reductions to be matched by sequences of instantaneous reductions, and timed reductions to be matched by timed reductions, possibly preceded and followed by sequences of instantaneous ones. To prove this property we will need the following technical result, which will also be used later:

Lemma 4.14. Suppose $\Gamma_1 \triangleright W_1 \mid T \simeq \Gamma_2 \triangleright W_2 \mid T$ where each channel occurring free in T does not occur free in W_1, nor in W_2 and is idle in both Γ_1 and Γ_2; then $\Gamma_1 \triangleright W_1 \simeq \Gamma_2 \triangleright W_2$.

Proof. See the Appendix, Page 55, for an outline. □

Proposition 4.15. Let $\Gamma_1 \triangleright W_1, \Gamma_2 \triangleright W_2$ be two well-formed configurations such that $\Gamma_1 \triangleright W_1 \simeq \Gamma_2 \triangleright W_2$. Then

(i) whenever $\Gamma_1 \triangleright W_1 \rightarrow\!\triangleright_i \Gamma_1' \triangleright W_1'$ there exists a configuration $\Gamma_2' \triangleright W_2'$ such that $\Gamma_2 \triangleright W_2 \rightarrow\!\triangleright_i^* \Gamma_2' \triangleright W_2'$, and $\Gamma_1' \triangleright W_1' \simeq \Gamma_2' \triangleright W_2'$,
(ii) whenever $\Gamma_1 \triangleright W_1 \rightarrow\!\triangleright_\sigma \Gamma_1' \triangleright W_1'$ there exists a configuration $\Gamma_2' \triangleright W_2'$ such that $\Gamma_2 \triangleright W_2 \rightarrow\!\triangleright_i^* \rightarrow\!\triangleright_\sigma \rightarrow\!\triangleright_i^* \Gamma_2' \triangleright W_2'$, and $\Gamma_1' \triangleright W_1' \simeq \Gamma_2' \triangleright W_2'$.

Proof. See the Appendix, Page 56. □

4.2.2. *Proving Completeness.* We are now in the position to prove that, for well-formed configurations, our proof methodology is also complete. Given two well-formed configurations $C_1 \simeq C_2$, there exists a bisimulation $\mathcal{S}$ such that $C_1 \mathcal{S} C_2$.

To prove completeness, we show that reduction barbed congruence is a bisimulation. That is, we need to show that for any extensional action α, if $C_1 \simeq C_2$ and $C_1 \stackrel{\alpha}{\longmapsto} C_1'$, then there exists C_2' such that $C_2 \stackrel{\hat{\alpha}}{\Longmapsto} C_2'$ and $C_1' \simeq C_2'$. The special cases $\alpha = \tau$ and $\alpha = \sigma$ follow as a direct consequence of Proposition 4.15. However, we state the results for the sake of consistency.

Proposition 4.16. [Preserving extensional τs] Suppose $\Gamma_1 \triangleright W_1 \simeq \Gamma_2 \triangleright W_2$ and $\Gamma_1 \triangleright W_1 \stackrel{\tau}{\longmapsto} \Gamma_1' \triangleright W_1'$. Then $\Gamma_2 \triangleright W_2 \Longmapsto \Gamma_2' \triangleright W_2'$ such that $\Gamma_1' \triangleright W_1' \simeq \Gamma_2' \triangleright W_2'$. □

Proposition 4.17. [Preserving extensional σs] Suppose $\Gamma_1 \triangleright W_1 \simeq \Gamma_2 \triangleright W_2$. Then $\Gamma_1 \triangleright W_1 \stackrel{\sigma}{\longmapsto} \Gamma_1' \triangleright W_1'$ implies $\Gamma_2 \triangleright W_2 \stackrel{\sigma}{\Longmapsto} \Gamma_2' \triangleright W_2'$ such that $\Gamma_1' \triangleright W_1' \simeq \Gamma_2' \triangleright W_2'$. □

Let us turn our attention to the remaining cases $\alpha \in \{c?v, \iota(c), \gamma(c,v)\}$. For each of them we define a distinguishing context T_α; these are defined so that, given a well-formed configuration C, $C \stackrel{\hat{\alpha}}{\Longmapsto} C'$ if and only if $C \mid T_\alpha \rightarrow^* C' \mid T_\alpha^\checkmark$, where $T_\alpha^\checkmark$ is uniquely determined by the action α. Intuitively, the latter corresponds to the first state reached by the testing component when it has detected that the configuration C has performed a weak α-action; the system $T_\alpha^\checkmark$ is called the successful state for the action α.

The tests T_α are defined below; here we assume that *eureka*, *fail* are fresh channels, while $\delta_{ok} = \delta_{no} = 1$.

$$
\begin{array}{rcl}
T_{\gamma(c,v)} & \stackrel{\text{def}}{=} & \nu d{:}(0,\cdot).(c[x].([x{=}v]d!\langle ok\rangle, \mathsf{nil}) + \mathit{fail}!\langle no\rangle \mid \sigma^2.[\exp(d)]\mathit{eureka}!\langle ok\rangle, \mathsf{nil}) \\
T_{c?v} & \stackrel{\text{def}}{=} & (c\,!\langle v\rangle.\mathit{eureka}!\langle ok\rangle + \mathit{fail}!\langle no\rangle) \\
T_{\iota(c)} & \stackrel{\text{def}}{=} & ([\exp(c)]\mathsf{nil}, \mathit{eureka}!\langle ok\rangle) + \mathit{fail}!\langle no\rangle.
\end{array}
$$

We also list their respective successful states $T_\alpha^\checkmark$:

$$
\begin{array}{rcl}
T^\checkmark_{\gamma(c,v)} & \stackrel{\text{def}}{=} & \nu d{:}(0,\cdot).(\sigma.d!\langle ok\rangle \mathsf{nil} \mid \sigma.[\exp(d)]\mathit{eureka}!\langle ok\rangle, \mathsf{nil}) \\
T^\checkmark_{c?v} & \stackrel{\text{def}}{=} & (\sigma^{\delta_v}.\mathit{eureka}!\langle ok\rangle) \\
T^\checkmark_{\iota(c)} & \stackrel{\text{def}}{=} & \sigma.\mathit{eureka}!\langle ok\rangle
\end{array}
$$

As an example we consider in detail the behaviour of the testing context $T_{\gamma(c,v)}$. This is designed to detect whether a configuration $\Gamma \triangleright W$ has performed a weak $\gamma(c,v)$-action. Let us discuss informally how the testing context $T_{\gamma(c,v)}$ operates. The fresh channels *eureka*, *fail* play a different role: *fail* ensures that the reception along channel c has finished, while *eureka* guarantees that the received values is actually v.

We provide a possible evolution of the testing contexts $T_\gamma(c,v)$ when running in a channel environment Γ such that $\Gamma(c) = (1, v)$, and then we discuss how it works.

$$
\begin{array}{lll}
 & \Gamma \triangleright T_{\gamma(c,v)} & \\
\rightarrow_\sigma & \Gamma_1 \triangleright T_1 = & \Gamma_1' \triangleright \nu d{:}(0,\cdot).(([v{=}v]d!\langle ok\rangle, \mathsf{nil}) + \mathit{fail}!\langle no\rangle \mid \\
 & & \mid \sigma.[\exp(d)]\mathit{eureka}!\langle ok\rangle, \mathsf{nil}) \\
\rightarrow_i & \Gamma^\checkmark \triangleright T^\checkmark = & \Gamma_2 \triangleright \nu d{:}(0,\cdot).(\sigma.d!\langle ok\rangle \mid \sigma.[\exp(d)]\mathit{eureka}!\langle ok\rangle, \mathsf{nil}) \\
\rightarrow_\sigma & \Gamma_3 \triangleright T_3 = & \Gamma_3 \triangleright \nu d{:}(0,\cdot).(d!\langle ok\rangle \mid [\exp(d)]\mathit{eureka}!\langle ok\rangle, \mathsf{nil}) \\
\rightarrow_i & \Gamma_4 \triangleright T_4 = & \Gamma_4 \triangleright \nu d{:}(1, ok).(\sigma \mid [\exp(d)]\mathit{eureka}!\langle ok\rangle, \mathsf{nil}) \\
\rightarrow_i & \Gamma_5 \triangleright T_5 = & \Gamma_5 \triangleright \nu d{:}(1, ok).(\sigma \mid \sigma.\mathit{eureka}!\langle ok\rangle) \\
\rightarrow_\sigma & \Gamma_6 \triangleright T_6 = & \Gamma_6 \triangleright \nu d{:}(0,\cdot).(\mathsf{nil} \mid \mathit{eureka}!\langle ok\rangle)
\end{array}
$$

Initially a configuration of the form $\Gamma \triangleright W \mid T_{\gamma(c,v)}$ has a weak barb at channel *fail*. Further, the testing component has an active receiver over channel c; note that the configuration $\Gamma \triangleright W \mid T_{\gamma(c,v)}$ is well-formed only if $\Gamma \vdash c : \mathbf{exp}$. If $\Gamma \triangleright W \mid T_{\gamma(c,v)} \stackrel{\gamma(c,v)}{\Longmapsto} \Gamma_1 \triangleright W'$, that is if $\Gamma(c) = (1, v)$, then after time passes the reception along channel c in the testing component $T_{\gamma(c,v)}$ terminates. Formally, we have the sequence of reductions $\Gamma \triangleright W \mid T_{\gamma(c,v)} \rightarrow_i^* \rightarrow_\sigma \rightarrow_i^* \Gamma_1 \triangleright W' \mid T_1$. Note that the component T_1 compares the received value along channel c with v; this test can only succeed, and as a consequence we obtain a further instantaneous reduction $\Gamma_1 \triangleright W' \mid T_1 \rightarrow_i \Gamma^\checkmark \triangleright W' \mid T^\checkmark$; In practice here we have $\Gamma^\checkmark = \Gamma_1$). At this point we have detected that the configuration $\Gamma_1 \triangleright W_1$ has performed the weak $\gamma(c,v)$-action, ending in $\Gamma_1 \triangleright W'$. The rest of the computation is already determined, at least for

the part concerning the testing component T_1, and leads $\Gamma^{\checkmark} \triangleright W' \mid T^{\checkmark}$ to output a barb on *eureka*; further, in this configuration it is not possible to output a barb on *fail* anymore.

To see why this is true, note that in $\Gamma^{\checkmark} \triangleright W' \mid T^{\checkmark}$ the testing component $T^{\checkmark}$ is waiting for time to pass, before broadcasting value *ok* along a restricted channel d. Formally, we have the sequence of reductions $\Gamma^{\checkmark} \triangleright W' \mid T^{\checkmark} \rightarrow_i^* \rightarrow_\sigma \Gamma_3 \triangleright W_3 \mid T_3 \rightarrow_i \Gamma_4 \triangleright W_4 \mid T_4$, where $\Gamma^{\checkmark} \triangleright W' \rightarrow_i^* \Gamma_2 \triangleright W_2$ and $W_3 = W_4$ (note that each instantaneous reduction performed by the tested component does not affect the test at this point).

Finally, in $\Gamma_4 \triangleright W_4 \mid T_4$ the test checks whether the restricted channel d is exposed. As this channel is effectively restricted in T_4, the test can only succeed, leading to $\Gamma_4 \triangleright W_4 \mid T_4 \rightarrow_i \Gamma_5 \triangleright W_5 \mid T_5$, where $\Gamma_5 = \Gamma_4$ and $W_5 = W_4$. At this point we can let time pass, via a sequence of reductions of the form $\Gamma_5 \triangleright W_5 \mid T_5 \rightarrow_i^* \rightarrow_\sigma \rightarrow_i^* \Gamma_6 \triangleright W_6 \mid T_6$. Now it is trivial to see that this configuration has a barb on *eureka*.

Note that in the computation of $\Gamma \triangleright W \mid T_{\gamma(c,v)}$ discussed above, there are two crucial checks that lead to enabling a barb over channel *eureka*:

- The received value is exactly v,
- The check that a broadcast along the restricted channel d is performed after two time instants. Since the broadcast along channel d is performed only one time instant after value v has been delivered, this check ensures that such a value has been actually delivered after one time instant.

Proposition 4.18. [Detecting Inputs] For any well-formed configuration $\Gamma \triangleright W$ we have that $\Gamma \triangleright W \overset{c?v}{\Longmapsto} \Gamma' \triangleright W'$ if and only if $\Gamma \triangleright W \mid T_{c?v} \rightarrow_i^* \Gamma' \triangleright W' \mid T^{\checkmark}_{c?v}$.

Proof. See the Appendix, Page 57. □

Proposition 4.19. [Detecting Exposure Checks] For any well-formed configuration $\Gamma \triangleright W$ we have that $\Gamma \triangleright W \overset{\iota(c)}{\Longmapsto} \Gamma' \triangleright W'$ if and only if $\Gamma \triangleright W \mid T_{\iota(c)} \rightarrow_i^* \Gamma' \triangleright W' \mid T^{\checkmark}_{\iota(c)}$.

Proof. See the Appendix, Page 58. □

Proposition 4.20. [Detecting Delivery of Values] For any well-formed configuration $\Gamma \triangleright W$ we have that $\Gamma \triangleright W \overset{\gamma(c,v)}{\Longmapsto} \Gamma' \triangleright W'$ if and only if $\Gamma \triangleright W \mid T_{\gamma(c,v)} \rightarrow_i^* \rightarrow \sigma \rightarrow_i^* \Gamma' \triangleright W' \mid T^{\checkmark}_{\gamma(c,v)}$.

Proof. See the Appendix, Page 59. □

Note that in Propositions 4.18, 4.19 and 4.20, we emphasized whether the reductions needed to reach the successful configuration $\Gamma \triangleright W' \mid T^{\checkmark}_{\alpha}$ from $\Gamma \triangleright W \mid T_\alpha$ are instantaneous or timed.

We have stated all the results needed to prove completeness.

Theorem 4.21. [Completeness] On well-defined configurations, reduction barbed congruence implies bisimilarity.

Proof. It is sufficient to show that the relation

$$\mathcal{S} \overset{\text{def}}{=} \{(\Gamma_1 \triangleright W_1 \,,\, \Gamma_2 \triangleright W_2) : \;\; \Gamma_1 \triangleright W_1 \simeq \Gamma_2 \triangleright W_2\}$$

is a bisimulation. To do so, suppose that $\Gamma_1 \triangleright W_1 \rightarrow \Gamma_2 \triangleright W_2$, and that $\Gamma_1 \triangleright W_1 \simeq \Gamma_2 \triangleright W_2$. If $\alpha = \tau$ or $\alpha = \sigma$, the result follows directly from propositions 4.16 and 4.17, respectively.

Now suppose that $\alpha = \gamma(c, v)$ for some channel c and value v. Let $\Gamma_1 \triangleright W_1 \overset{\gamma(c,v)}{\longmapsto} \Gamma_1' \triangleright W_1'$; by Proposition 4.20 it follows that $\Gamma_1 \triangleright W_1 \mid T_{\gamma(c,v)} \rightarrow_i^* \rightarrow_\sigma \rightarrow_i^* \Gamma_1' \triangleright W_1' \mid T^{\checkmark}_{\gamma(c,v)}$. By the contextuality of reduction barbed congruence, and by Proposition 4.15, it follows that $\Gamma_1 \triangleright W_2 \mid T_{\gamma(c,v)} \rightarrow_i^* \rightarrow_\sigma \rightarrow_i^* C_2$

for some C_2 such that $\Gamma_1' \triangleright W_1' \mid T^{\surd}_{\gamma(c,v)} \simeq C_2$. Let $C_2 = \Gamma_2' \triangleright \widehat{W_2}$; note that $\Gamma_1' \vdash \mathit{eureka} : \mathbf{idle}$ (recall that we assumed that *eureka* is a fresh channel), so that by Proposition 4.13 it follows that $\Gamma_2' \vdash \mathit{eureka} : \mathbf{idle}$. Further, $\Gamma_1' \triangleright W_1' \mid T^{\surd}_{\gamma(c,v)} \Downarrow_{\mathit{eureka}}$ and $\Gamma_1' \triangleright W_1' T^{\surd}_{\gamma(c,v)} \not\Downarrow_{\mathit{fail}}$; therefore, we also have that $\Gamma_2' \triangleright \hat{W}_2 \Downarrow_{\mathit{eureka}}$ and $\Gamma_2' \triangleright \hat{W}_2 \not\Downarrow_{\mathit{fail}}$. Now, by inspecting all the possible evolutions of the configuration $\Gamma_2 \triangleright W_2 \mid T_{\gamma(c,v)}$ it follows that the sequence of reductions $\Gamma_1 \triangleright W_2 \mid T_{\gamma(c,v)} \rightarrow\!\triangleright^*_i \rightarrow\!\triangleright_\sigma \rightarrow\!\triangleright^*_i \Gamma_2' \triangleright \hat{W}_2$, where $\Gamma_2' \vdash \mathit{eureka} : \mathbf{idle}$, $\Gamma_2' \triangleright \hat{W}_2 \Downarrow_{\mathit{eureka}}$ and $\Gamma_2' \triangleright \hat{W}_2 \not\Downarrow_{\mathit{fail}}$, is possible only if $\hat{W}_2 = W_2' \mid T^{\surd}_{\gamma(c,v)}$. Consequently, Proposition 4.20 ensures that $\Gamma_2 \triangleright W_2 \overset{\gamma(c,v)}{\Longmapsto} \Gamma_2' \triangleright W_2'$.

We also need to show that $\Gamma_1' \triangleright W_1' \simeq \Gamma_2' \triangleright W_2'$; but this follows immediately from Lemma 4.14 and the fact that $\Gamma_1' \triangleright W_1' \mid T^{\surd}_{\gamma(c,v)} \simeq \Gamma_2' \triangleright W_2' \mid T^{\surd}_{\gamma(c,v)}$.

It remains to check the cases $\alpha = c?v$ and $\alpha = \iota(c)$; these can be proved analogously to the previous case, using proposition 4.18 and 4.19, respectively, in lieu of Proposition 4.20. □

5. Applications

In this section, we show how our calculus CCCP can be used to model different interesting behaviours which arise at the MAC sub-layer [26] of wireless networks. Then, we exploit our bisimulation proof technique to provide examples of behaviourally equivalent networks. In particular we give some examples comparing the behaviour of routing protocols and *Time Division Multiplexing*.

We start with some simple examples. The first show that stations which do not transmit on unrestricted channels can not be detected. To this end we use $\mathrm{fsn}(W)$ to denote the set of unrestricted channel names in the code W which have transmission occurrences. Formally $\mathrm{fsn}(W)$ is defined inductively on (a possibly open system term) W as the least set such that

$$\begin{aligned}
\mathrm{fsn}(\mathsf{nil}) &= \mathrm{fsn}(X) = \emptyset \\
\mathrm{fsn}(!\langle c\rangle.\nu P) &= \{c\} \cup \mathrm{fsn}(P) \\
\mathrm{fsn}(\tau.P) &= \mathrm{fsn}(\sigma.P) = \mathrm{fsn}(c[x].P) = \mathrm{fsn}(\mathsf{fix}\, X.P) = \mathrm{fsn}(P) \\
\mathrm{fsn}(P + Q) &= \mathrm{fsn}([b]P, Q) = \mathrm{fsn}(\lfloor c?(x).P \rfloor Q) = \mathrm{fsn}(P) \cup \mathrm{fsn}(Q)
\end{aligned}$$

$$\begin{aligned}
\mathrm{fsn}(W_1 \mid W_2) &= \mathrm{fsn}(W_1) \cup \mathrm{fsn}(W_2) \\
\mathrm{fsn}(\nu c : (n, v).W) &= \mathrm{fsn}(W_1) \setminus \{c\}
\end{aligned}$$

Example 5.1. [Unobservable systems] Consider a wireless system in which no station can broadcast on any free channel. Intuitively none of its behaviour should be observable. In CCCP this means that the system should be behaviourally equivalent to the *empty* system nil. f

Formally consider the configuration $\Gamma \triangleright \mathsf{nil}$ where Γ is an arbitrary channel environment. This configuration has non-trivial extensional behaviour. For example it is input enabled, and so can perform all extensional actions of the form $c?v$. It can also perform σ actions, indicating the passage of time.

Now let W be arbitrary station code such that $\mathrm{fsn}(W) = \emptyset$, that is it can not broadcast on any free channel. The configuration $\Gamma \triangleright W$ has similar behaviour. Indeed let $\mathcal{S}$ be the relation

$$\{(\Gamma \triangleright W, \Gamma \triangleright \mathsf{nil}) \mid \mathrm{fsn}(W) = \emptyset\}$$

Then it is straightforward to show that $\mathcal{S}$ is a strong bisimulation in the extensional LTS. Our soundness result therefore ensures that

$$\Gamma \triangleright W \simeq \Gamma \triangleright \mathsf{nil}$$

whenever $\mathrm{fsn}(W) = 0$. □

Next we consider what happens when a channel becomes permanently exposed. This situation can be modelled by using two stations s_0, s_1 which repeatedly send a value along channel c; each broadcast performed by s_1 takes place before the transmission of s_0 ends, and vice versa. In this case we say that the channel c is *corrupted*. Clearly, if a system transmits only on corrupted channels; then it cannot be detected. Let us see how this scenario is reflected in our behavioural theory.

Example 5.2. [Noise obfuscates transmissions] Let v be a value such that $\delta_v = 2$ and let $\mathrm{Snd}(c)$ denote the code $\mathsf{fix}\, X.c\,!\langle v\rangle.X$, which continually broadcasts an arbitrary value v along c. To model the two stations s_0 and s_1 discussed informally above we use the code $\mathrm{Noise}(c) = \mathrm{Snd}(c) \mid \sigma.\mathrm{Snd}(c)$.

Then, consider a configuration $\Gamma \triangleright W$ such that $\mathrm{fsn}(W) \subseteq \{c\}$; that is does not transmit on free channels different from c. Then

$$\Gamma \triangleright W \mid \mathrm{Noise(c)} \;\simeq\; \Gamma \triangleright \mathrm{Noise(c)}$$

To prove this, it is sufficient to exhibit bisimulation containing the pair of configurations $(\Gamma \triangleright W \mid \mathrm{Noise(c)},\ \Gamma \triangleright \mathrm{Noise(c)})$.

We use the following abbreviations:

$$\mathrm{Noise}'(c) = \sigma^2.\mathrm{Snd}(c) \mid \sigma.\mathrm{Snd}(c)$$
$$\mathrm{Noise}''(c) = \sigma.\mathrm{Snd}(c) \mid \mathrm{Snd}(c)$$
$$\mathrm{Noise}'''(c) = \sigma.\mathrm{Snd}(c) \mid \sigma^2.\mathrm{Snd}(c)$$

Then let $\mathcal{S}$ denote the following set of pairs of configurations:

$$\begin{array}{l}\{(\Delta \triangleright W \mid \mathrm{Noise(c)} \quad , \quad \Delta' \triangleright \mathrm{Noise(c)}),\\ (\Delta \triangleright W \mid \mathrm{Noise}'(c) \quad , \quad \Delta' \triangleright \mathrm{Noise}'(c)),\\ (\Delta \triangleright W \mid \mathrm{Noise"}(c) \quad , \quad \Delta' \triangleright \mathrm{Noise}''(c)),\\ (\Delta \triangleright W \mid \mathrm{Noise"'}(c) \quad , \quad \Delta' \triangleright \mathrm{Noise}'''(c)) \quad \mid\\ \qquad \Delta, \Delta' \vdash c : \mathbf{exp}, \mathrm{fsn}(W) \subseteq \{c\}\ \}\end{array}$$

Then it is possible to check that $\mathcal{S}$ is a weak bisimulation in the extensional LTS. At least intuitively, this is because in the extensional LTS all outputs fired along the obfuscated channel c corresponds to internal actions; further, in the configurations included in $\mathcal{S}$, channel c is never released, so that neither $\iota(c)$-actions nor $\gamma(c, v)$-actions can be performed by any configuration included in $\mathcal{S}$. □

The *Carrier Sense Multiple Access* (CSMA) scheme [24] is a widely used MAC-layer protocol in which a device senses the channel (*physical carrier sense*) before transmitting. More precisely, if the channel is sensed free the sender starts transmitting immediately, that is in the next instant of time[5]; if the channel is busy, that is some other station is transmitting, the device keeps listening to the channel until it becomes idle and then starts transmitting immediately. This strategy is called *1-persistent* CSMA and can be easily expressed in our calculus in terms of the following process:

$$c!!\langle v\rangle.P = \mathsf{fix}\, X.[\exp(c)]X, c\,!\langle v\rangle.P$$

So, by definition CSMA transmissions are delayed whenever the channel is busy.

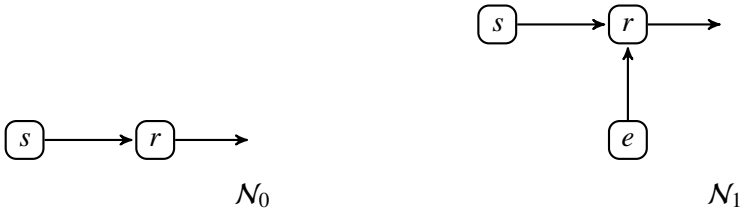
In the next example we prove a natural property of CSMA transmissions.

Example 5.3. [Delay in CSMA broadcast] Suppose $\Gamma \vdash_{\mathrm{t}} c : n$ for some $n > 0$. Then, for any $k \leq n + 1$

$$\Gamma \triangleright c!!\langle v\rangle.P \;\simeq\; \Gamma \triangleright \sigma^k.c!!\langle v\rangle.P \tag{5.1}$$

[5]Recall that in wireless systems channels are half-duplex.

Table 9 A simple topology for a network



.

Intuitively, since $\Gamma \vdash_{\mathrm{t}} = n$, the transmission of value v in $\Gamma \triangleright c!!\langle v\rangle.P$ can take place only after at least n instants of time. The same happens in $\Gamma \triangleright \sigma^k.c!!\langle v\rangle.P$.

Formally, to prove (5.1) we need to exhibit a bisimulation $\mathcal{S}$ which contains all pairs of the form $(\Gamma \triangleright c!!\langle v\rangle.P,\ \sigma^k.c!!\langle v\rangle.P)$, where Γ is such that $\Gamma \vdash_{\mathrm{t}} n > 0$ for some n satisfying $k \leq (n+1)$. One possible $\mathcal{S}$ takes the form $\mathcal{R} \cup \mathcal{I}d$ where $\mathcal{I}d$ is the identity relation over configurations and $\mathcal{R}$ is given by:

$$\mathcal{R} = \{(\Delta_n \triangleright c!!\langle v\rangle.P, \Delta_n \triangleright \sigma^h.c!!\langle v\rangle.P) \ \mid \ \Delta_n \vdash_{\mathrm{t}} c : n, h \leq n\}$$

□

In our calculus the network topology is *assumed to be* flat. However, we can exploit the presence of multiple channels to model networks with a more complicated topological structure. The idea is to associate a particular channel with a collection of stations which are in the same neighbourhood.

Example 5.4. [Network Topology] Suppose that we want to model a network with two stations s, r with the following features:

- the range of transmission of s is too short to reach external agents,
- the station r is in the range of transmission of s,
- the range of transmission of r is long enough to also reach external agents.

A graphical representation of the network we want to model is given as $\mathcal{N}_0$ of Table 9. We can model this network topology by using a specific restricted channel, say d, for the local communication between stations s and r. In CCCP a wireless system running on $\mathcal{N}_0$ would therefore take the form

$$C_0 = \Gamma \triangleright \nu d : (0, \cdot).(S \mid R)$$

where

- S represents the code running at station s; it can therefore only broadcast and receive along the restricted channel d (recall that we do not want station s to be able to communicate directly with the external environment)
- R represents the code running at station r; it can only receive values along the restricted channel d (since in $\mathcal{N}_0$ station r can receive messages broadcast by station r, but not by the external environment), while it is free to broadcast on other channels (since station r is able to broadcast messages to the external environment)

As a specific example we could let S denote the single broadcast $d!\langle v\rangle$, and $R = \mathsf{fix}\, X.\lfloor d?(x).c!\langle x\rangle\rfloor X$. Then in the configuration C_0 the station s broadcasts as a value and station r acts as a forwarder; this behaviour is reminiscent of range repeaters in wireless terminology.

Suppose now that we want to add a second station e to the above network topology, so that

- broadcasts from e can be detected by r; this can be accomplished by allowing the process used to model station e to broadcasts along a restricted channel d.
- broadcasts from e can not reach s, nor the external environment. For this to be true, it is sufficient to require that the process which models the behaviour of station e can broadcast values only along the restricted channel d; further, in order for ensuring that the station e cannot detect values broadcast by s, we require that the process used to represent station e does not use receivers along channel d.

The network topology we wish to model is depicted as $\mathcal{N}_1$ in Table 9 and so a wireless system running on this network takes the form

$$C_1 = \nu d{:}(0,\cdot).(S \mid R \mid E)$$

where E is the code running at station e. As an example we could take E to be the faulty code $d!\langle v\rangle + \tau.\mathsf{nil}$.

Then in C_1 station r still acts as a forwarder for station s; however station e can non-deterministically decide whether to corrupt the transmission from node s to r, causing a collision.

Let us assume that the transmission time of the value used in these networks, v, satisfies $\delta_v = \delta_{\mathsf{err}}$. Then we can show

$$C_0 \simeq \Gamma \triangleright \sigma^{\delta_v}.c!\langle v\rangle$$
$$C_1 \simeq \Gamma \triangleright \tau.\sigma^{\delta_v}.c!\langle v\rangle + \tau.\sigma^{\delta_v}.c!\langle \mathsf{err}\rangle$$

Intuitively the reasons for these equivalences are obvious. The transmission along channel d is restricted in C_0, so it cannot be observed by the external environment. The only activity which can be observed is the broadcast of value v along channel c, which takes place after δ_v instants of time. For C_1, a collision can happen along channel d, which is again restricted; the only activity that can be detected by the external environment is a transmission which takes place after δ_v instants of time. Such a transmission will contain either the value v or an error message of length δ_v.

The formal proof of these identities involves exhibiting two bisimulations, containing the relevant pairs of configurations. Here we exhibit a bisimulation for showing that $C_1 \simeq \Gamma \triangleright \tau.\sigma^{\delta_v}.c!\langle v\rangle$. For the sake of simplicity, let $\delta_{\mathsf{err}} = \delta_v = 1$ and define the system terms

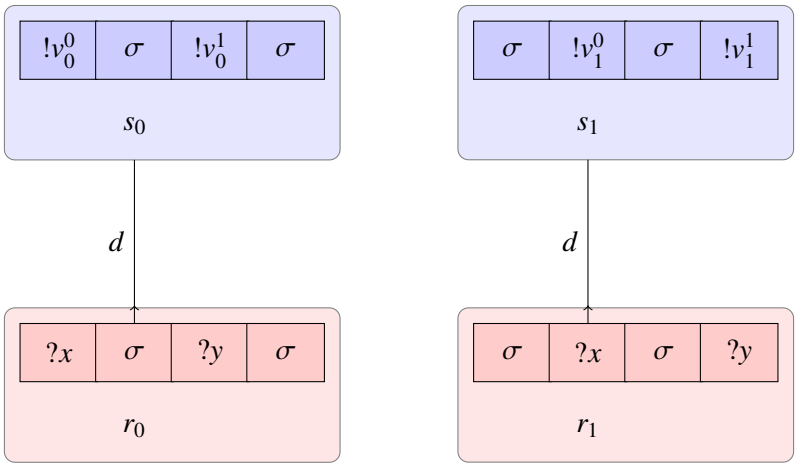
$$\begin{array}{lcllcl}
W & = & \nu d : (0,\cdot).(S \mid E \mid R) & \qquad W_s & = & \nu d : (1,v).(\sigma \mid E \mid c[x].c!\langle x\rangle)\\
W_e & = & \nu d : (1,\mathsf{err}).(S \mid \sigma \mid c[x].c!\langle x\rangle) & W' & = & \nu d : (0,\cdot).(S \mid \mathsf{nil} \mid R)\\
W'' & = & \nu d : (1,\mathsf{err}).(\sigma \mid \sigma \mid c[x].c!\langle x\rangle) & W_{\mathrm{ok}} & = & \nu d : (0,\cdot).(\mathsf{nil} \mid \mathsf{nil} \mid c!\langle v\rangle)\\
W_{\mathsf{err}} & = & \nu d : (0,\cdot).(\mathsf{nil} \mid \mathsf{nil} \mid c!\langle \mathsf{err}\rangle) & W_c & = & \nu d : (0,\cdot).(\mathsf{nil} \mid \mathsf{nil} \mid \sigma)
\end{array}$$

Then it is easy to show that the relation

$$\begin{array}{rcll}
\mathcal{S} = \{ & (\Delta \triangleright W & , & \Delta \triangleright \tau.\sigma.c!\langle v\rangle + \tau.\sigma.c!\langle \mathsf{err}\rangle) \quad ,\\
& (\Delta \triangleright W_s & , & \Delta \triangleright \tau.\sigma.c!\langle v\rangle + \tau.\sigma.c!\langle \mathsf{err}\rangle) \quad ,\\
& (\Delta \triangleright W_e & , & \Delta \triangleright \sigma.c!\langle \mathsf{err}\rangle) \quad ,\\
& (\Delta \triangleright W' & , & \Delta \triangleright \sigma.c!\langle v\rangle) \quad ,\\
& (\Delta \triangleright W'' & , & \Delta \triangleright \sigma.c!\langle \mathsf{err}\rangle) \quad ,\\
& (\Delta \triangleright W_{\mathrm{ok}} & , & \Delta \triangleright c!\langle v\rangle) \quad ,\\
& (\Delta \triangleright W_{\mathsf{err}} & , & \Delta \triangleright c!\langle \mathsf{err}\rangle) \quad ,\\
& (\Delta \triangleright W_c & , & \Delta \triangleright \sigma) \\
\mid & \Delta & & \text{arbitrary channel environment} \quad \}
\end{array}$$

is a weak bisimulation. □

Table 10 Two transmitting stations using different time slots to broadcast values



The next example shows how the TDMA modulation technique [52] can be described in CCCP. *Time Division Multiple Access* (TDMA) is a type of Time Division Multiplexing, where instead of having one transmitter connected to one receiver, there are multiple transmitters. TDMA is used in the digital 2G cellular systems such as *Global System for Mobile Communications* (GSM). TDMA allows several users to share the same frequency channel by dividing the signal into different time slots. The users transmit in rapid succession, one after the other, each using his own time slot. This allows multiple stations to share the same transmission medium (e.g. radio frequency channel) while using only a part of its channel capacity.

As a simple example let us describe how two messages v_0 and v_1 can be delivered in TDMA style; for simplicity, we assume $\delta_{v_0} = \delta_{v_1} = 2$. The main idea here is to split each of these values into two packets of length one, transmit the packets individually, which will then be concatenated together before being forwarded to the external environment. So let us assume values $v_0^0, v_0^1, v_1^0, v_1^1$, each of which requires one time instant to be transmitted, and a binary operator $\circ$ for composing messages such that

$$\begin{aligned} v_0^0 \circ v_0^1 &= v_0 \\ v_1^0 \circ v_1^1 &= v_1 \\ v \circ \mathsf{err} &= \mathsf{err} \circ v = \mathsf{err} \end{aligned}$$

where v is an arbitrary value; in this case we assume that $\delta_{\mathsf{err}} = 2$.

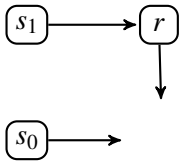
More specifically, for this example we assume four different stations, s_0, s_1, r_0, r_1, running the code $\hat{S}_0$, $\hat{S}_1$, $\hat{R}_0$, $\hat{R}_1$ respectively. The network we consider for modelling the TDMA transmission is then given by

$$C_0 = \Gamma \triangleright \nu d{:}(0,\cdot)(\hat{S}_0 \mid \hat{S}_1 \mid \hat{R}_0 \mid \hat{R}_1)$$

where

$$\begin{aligned} \hat{S}_0 &= d\,!\langle v_0^0\rangle.\sigma.d!\langle v_0^1\rangle \\ \hat{S}_1 &= \sigma.d\,!\langle v_1^0\rangle.\sigma.d!\langle v_1^1\rangle \\ \hat{R}_0 &= \lfloor d?(x).\sigma.\lfloor d?(y).\sigma.c!\langle x \circ y\rangle\rfloor\rfloor \\ \hat{R}_1 &= \sigma.\lfloor d?(x).\sigma.\lfloor d?(y).\sigma^2.c!\langle x \circ y\rangle\rfloor\rfloor \end{aligned}$$

Table 11 Forwarding two messages to the external environment



The intuitive behaviour of this network is depicted in Table 10. Station s_0 wishes to broadcast value v_0, while s_1 wishes to broadcast value v_1. They both use the same (restricted) channel d to broadcast their respective values; however, both stations split the value to be broadcast in two packets. Value v_0 is split in v_0^0 and v_0^1, while v_1 is split in v_1^0 and v_1^1.

The two stations run a TDMA protocol with a time frame of length two. Station s_0 takes control of the first time frame, hence transmitting its two packets v_0^0 and v_0^1 in the first and the third time slot, respectively. Station s_1 takes control of the second time frame; hence the two packets v_1^0 and v_1^1 are broadcast in the second and fourth time slot, respectively.

Stations r_0 and r_1 wait to collect the values broadcast along channel d. However, the former is interested only in packets sent in the first time frame, while the latter detects only values sent in the second time frame. At the end of their associated time frame the stations r_0 and r_1 have received two packets which are concatenated together and then broadcast to the external environment along channel c. Note that station r_1 is a little slower than r_0, for we have added a delay of two time units before broadcasting the concatenated values.

As an alternative to TDMA, the two values v_0, v_1 can be also be delivered to the external environment by means of a simple routing, along the lines suggested in Example 5.4. Here we consider the configuration

$$C_1 = \Gamma \triangleright \nu d{:}(0,\cdot).(S_0 \mid S_1 \mid R)$$

where

$$\begin{aligned} S_0 &= \sigma^4.c!\langle v_0\rangle \\ S_1 &= \sigma^4.d!\langle v_1\rangle \\ R &= d?(x).c!\langle x\rangle \end{aligned}$$

Intuitively, the configuration C_1 models three wireless stations s_0, s_1, r, running the code S_0, S_1, R, respectively, and connected as in Table 11. Station s_0 waits four instants of time, then it broadcasts value v_0 directly to the external environment via the free channel c. Similarly, after four instants of time the station s_1 broadcasts value v_1 to station r via the restricted channel d. Finally, r forwards the message to the external environment via the free channel c.

From the point of view of the external environment the configuration C_1 performs the following activities:

- it remains idle for the first four instants of time
- it transmits value v_0 in the fifth and sixth time instants
- it transmits value v_1 in the seventh and eighth time instants.

In this manner, at least informally the observable behaviour of C_1, which uses direct routing, is the same as that of C_0, which uses TDMA.

Formally, we can prove

$$C_0 \simeq C_1 \tag{5.2}$$

However, instead of proving this by giving a bisimulation containing this pair of configurations, we prove them individually bisimilar to a simpler specification. Let $\mathcal{S}_1$ denote the configuration $\Gamma \triangleright S_1$ where S_1 is the code

$$\sigma^4.c\,!\langle v_0\rangle.c\,!\langle v_1\rangle.$$

Then we can show that $C_0 \approx \mathcal{S}_1$ and $C_1 \approx \mathcal{S}_1$, from which (5.2) follows by soundness. Let us show that $C_0 \simeq \mathcal{S}_1$; for the sake of simplicity, it will be convenient to define the following system terms:

$$\begin{array}{lcrlcl}
\hat{S}_0^n & = & \sigma^n.d!\langle v_0^1\rangle & \hat{S}_1' & = & d\,!\langle v_1^0\rangle.\sigma.d!\langle v_1^1\rangle \\
\hat{S}_1^n & = & \sigma^n.d!\langle v_1^1\rangle & (\hat{R}_0)^{\mathrm{act}} & = & d[x].\sigma.\lfloor d?(y).\sigma.c!\langle x \circ y\rangle\rfloor \\
\hat{R}_0' & = & \lfloor d?(y).\sigma.c!\langle v_0^0 \circ y\rangle\rfloor & (\hat{R}_0')^{\mathrm{act}} & = & d[y].\sigma.c!\langle v_0^0 \circ y\rangle \\
\hat{R}_0^f & = & c!\langle v_0^0 \circ v_0^1\rangle & \hat{R}_1' & = & \lfloor d?(x).\sigma.\lfloor d?(y).\sigma^2.c!\langle x \circ y\rangle\rfloor\rfloor \\
(\hat{R}_1')^{\mathrm{act}} & = & d[x].\sigma.\lfloor d?(y).\sigma^2.c!\langle x \circ y\rangle\rfloor & \hat{R}_1'' & = & \lfloor d?(y).\sigma^2.c!\langle v_1^0 \circ y\rangle\rfloor \\
(\hat{R}_1'')^{\mathrm{act}} & = & d[y].sigma^2.c!\langle v_1^0 \circ y\rangle & \hat{R}_1^f & = & c!\langle v_1^0 \circ v_1^1\rangle \\
W_n & = & \sigma^n.c\,!\langle v_0\rangle.c!\langle v_1\rangle & & &
\end{array}$$

Then the relation

$$\begin{array}{rcll}
\mathcal{R} = \{ & (\Delta \triangleright \nu d : (0,\cdot).(\hat{S}_0 \mid \hat{S}_1 \mid \hat{R}_0 \mid \hat{R}_1) & , & \Delta \triangleright W_4) \quad , \\
& (\Delta \triangleright \nu d : (1,v_0^0).(\hat{S}_0^2 \mid \hat{S}_1 \mid \hat{R}_0^{\mathrm{act}} \mid \hat{R}_1) & , & \Delta \triangleright W_4) \quad , \\
& (\Delta \triangleright \nu d : (0,\cdot).(\hat{S}_0^1 \mid \hat{S}_1' \mid \sigma.\hat{R}_0' \mid \hat{R}_1') & , & \Delta \triangleright W_3) \quad , \\
& (\Delta \triangleright \nu d : (1,v_1^0).(\hat{S}_0^1 \mid \hat{S}_1^2 \mid \sigma.\hat{R}_0' \mid (\hat{R}_1')^{\mathrm{act}}) & , & \Delta \triangleright W_3) \quad , \\
& (\Delta \triangleright \nu d : (0,\cdot).(d!\langle v_0^1\rangle \mid \hat{S}_1^1 \mid \hat{R}_0' \mid \sigma.\hat{R}_1'') & , & \Delta \triangleright W_2) \quad , \\
& (\Delta \triangleright \nu d : (1,v_0^1).(\sigma \mid \hat{S}_1^1 \mid (\hat{R}_0')^{\mathrm{act}} \mid \sigma.\hat{R}_1'') & , & \Delta \triangleright W_2) \quad , \\
& (\Delta \triangleright \nu d : (0,\cdot).(\mathsf{nil} \mid c!\langle v_1^1\rangle \mid \sigma.\hat{R}_0^f \mid \hat{R}_1'') & , & \Delta \triangleright W_1) \quad , \\
& (\Delta \triangleright \nu d : (1,v_1^1).(\mathsf{nil} \mid c!\langle v_1^1\rangle \mid \sigma.\hat{R}_0^f \mid (\hat{R}_1'')^{\mathrm{act}}) & , & \Delta \triangleright W_1) \quad , \\
& (\Delta \triangleright \nu d : (0,\cdot).(\mathsf{nil} \mid \mathsf{nil} \mid c!\langle v_0\rangle \mid \sigma^2.c!\langle v_1\rangle) & , & \Delta \triangleright c\,!\langle v_0\rangle.c!\langle v_1\rangle) \quad , \\
& (\Delta \triangleright \nu d : (0,\cdot).(\mathsf{nil} \mid \mathsf{nil} \mid \sigma^2 \mid \sigma^2.c!\langle v_1\rangle) & , & \Delta \triangleright \sigma^2.c!\langle v_1\rangle) \quad , \\
& (\Delta \triangleright \nu d : (0,\cdot).(\mathsf{nil} \mid \mathsf{nil} \mid \sigma \mid \sigma.c!\langle v_1\rangle) & , & \Delta \triangleright \sigma.c!\langle v_1\rangle) \quad , \\
& (\Delta \triangleright \nu d : (0,\cdot).(\mathsf{nil} \mid \mathsf{nil} \mid \mathsf{nil} \mid c!\langle v_1\rangle) & , & \Delta \triangleright c!\langle v_1\rangle) \quad , \\
& (\Delta \triangleright \nu d : (0,\cdot).(\mathsf{nil} \mid \mathsf{nil} \mid \mathsf{nil} \mid \sigma^2) & , & \Delta \triangleright \sigma^2) \quad , \\
& (\Delta \triangleright \nu d : (0,\cdot).(\mathsf{nil} \mid \mathsf{nil} \mid \mathsf{nil} \mid \sigma) & , & \Delta \triangleright \sigma) \quad , \\
& (\Delta \triangleright \nu d : (0,\cdot).(\mathsf{nil} \mid \mathsf{nil} \mid \mathsf{nil} \mid \mathsf{nil}) & , & \Delta \triangleright \mathsf{nil}) \quad , \\
\mid & \Delta \text{ arbitrary channel environment} & & \}
\end{array}$$

is a bisimulation. Below we also show that $C_1 \simeq \mathcal{S}_1$; for the sake of simplicity, define the following terms:

$$\begin{array}{lcllcl}
S_0^n & = & \sigma^n.c!\langle v_0\rangle & S_1^n & = & \sigma^n.d!\langle v_1\rangle \\
R' & = & d[x].c!\langle x\rangle & W_n & = & \sigma^n.c\,!\langle v_0\rangle.c!\langle v_1\rangle
\end{array}$$

for any $n \in \mathbb{N}$. Then the relation

$$\begin{array}{rcll}
\mathcal{R}' = \{ & (\Delta \triangleright \nu d : (0,\cdot).(S_0^n \mid S_1^n \mid R) & , & \Delta \triangleright W_n) \quad , \\
& (\Delta \triangleright \nu d : (0,\cdot).(\sigma^2 \mid d!\langle v_1\rangle \mid R) & , & \Delta \triangleright \sigma^2.c!\langle v_1\rangle) \quad , \\
& (\Delta \triangleright \nu d : (2,v_1).(c!\langle v_0\rangle \mid \sigma^2 \mid R') & , & \Delta \triangleright c\,!\langle v_0\rangle.c!\langle v_1\rangle) \quad , \\
& (\Delta \triangleright \nu d : (2,v_1).(\sigma^2 \mid \sigma^2 \mid R') & , & \Delta \triangleright \sigma^2.c!\langle v_1\rangle) \quad , \\
& (\Delta \triangleright \nu d : (1,v_1).(\sigma \mid \sigma \mid R') & , & \Delta \triangleright \sigma.c!\langle v_1\rangle) \quad , \\
& (\Delta \triangleright \nu d : (0,\cdot).(\mathsf{nil} \mid \mathsf{nil} \mid c!\langle v_1\rangle) & , & \Delta \triangleright c!\langle v_1\rangle) \quad \mid \\
\mid & \Delta \text{ arbitrary channel environment} & & \}
\end{array}$$

is a relation which contains the most relevant couples needed for showing that $C_1 \approx \mathcal{S}_1$.

Example 5.5. As a final example we can modify the behaviour of the two configurations C_0 and C_1 seen above by adding the possibility of getting a *collision* when delivering values v_0, v_1 to the external environment. In the routing case, this is accomplished by requiring that both stations s_0, s_1 can either broadcast their value directly to the external environment or to the forwarder node r, while in the TDMA case it is sufficient to allow both the stations s_0, s_1 to non-deterministically choose the time slot to be used to broadcast packets.

To this end, let

$$\begin{array}{rcl} S_0^{\mathsf{c}} & = & \tau.\sigma^4.c!\langle v_0\rangle + \tau.\sigma^4.d!\langle v_0\rangle \\ S_1^{\mathsf{c}} & = & \tau.\sigma^4.c!\langle v_1\rangle + \tau.\sigma^4.d!\langle v_1\rangle \\ \hat{S}_0^{\mathsf{c}} & = & d\,!\langle v_0^0\rangle.\sigma.d!\langle v_0^1\rangle + \tau.\sigma.d\,!\langle v_0^0\rangle.\sigma.d!\langle v_0^1\rangle \\ \hat{S}_1^{\mathsf{c}} & = & d\,!\langle v_1^0\rangle.\sigma.d!\langle v_1^1\rangle + \tau.\sigma.d\,!\langle v_1^0\rangle.\sigma.d!\langle v_1^1\rangle \end{array}$$

and consider the configurations

$$\begin{array}{rcl} C_1^{\mathsf{c}} & = & \Gamma \triangleright \nu d{:}(0,\cdot).(S_0^{\mathsf{c}} \mid S_1^{\mathsf{c}} \mid R) \\ C_0^{\mathsf{c}} & = & \Gamma \triangleright \nu d{:}(0,\cdot).(\hat{S}_0^{\mathsf{c}} \mid \hat{S}_1^{\mathsf{c}} \mid \hat{R}_0 \mid \hat{R}_1) \end{array}$$

It is not difficult to see informally that the observable behaviour of these two configurations is the same. Specifically

- either value v_0 is broadcast in the fifth and sixth time slots and v_1 is broadcast in the seventh and eighth instants of time slots, or
- value v_1 is broadcast in the fifth and sixth time slots, while value v_0 is broadcast in the seventh and eighth time slots, or
- a collision occur in the fifth and sixth time slots, or
- a collision occur in the seventh and eighth time slots.

This informal behaviour can be described by the term

$$\begin{array}{rcll} S_2 & = & \tau.\sigma^4.c\,!\langle v_0\rangle.c!\langle v_1\rangle & + \\ & & \tau.\sigma^4.c\,!\langle v_1\rangle.c!\langle v_0\rangle & + \\ & & \tau.\sigma^4.c!\langle \mathsf{err}\rangle & + \\ & & \tau.\sigma^6.c!\langle \mathsf{err}\rangle & \end{array}$$

and once more we can exhibit bisimulations to establish $\Gamma \triangleright S_2 \approx C_0^{\mathsf{c}}$ and $\Gamma \triangleright S_2 \approx C_1^{\mathsf{c}}$. Then soundness again ensures that

$$C_0^{\mathsf{c}} \simeq C_1^{\mathsf{c}}$$

□

6. Conclusions and related work

In this paper we have given a behavioural theory of wireless systems at the MAC level. In our framework individual wireless stations broadcast information to their neighbours along virtual channels. These broadcasts take a certain amount of time to complete, and are subject to collisions. If a broadcast is successful a recipient may choose to ignore the information it contains, or may act on it, in turn generating further broadcasts. We believe that our reduction semantics, given in Section 2, captures much of the subtlety of intensional MAC-level behaviour of wireless systems.

Then based on this reduction semantics we defined a natural contextual equivalence between wireless systems which captures the intuitive idea that one system can be replaced by another in a larger network without affecting the observable behaviour of the original network. In the main result of the paper, we then gave a sound and complete characterisation of this behavioural equivalence in terms of *extensional actions*. This characterisation is important for two reasons. Firstly it gives an understanding of which aspects of the intensional behaviour is important from the point of view of external users of these wireless systems. Secondly it gives a powerful sound and complete co-inductive proof method for demonstrating that two systems are behaviourally equivalent. We have also demonstrated the viability of this proof methodology by a series of examples.

Let us now examine some relevant related work. We start with the literature on process calculi for wireless systems. Nanz and Hankin [37] have introduced the first (untimed) calculus for Mobile Wireless Networks (CBS$^\sharp$), relying on a graph representation of node localities. The main goal of that paper is to present a framework for specification and security analysis of communication protocols for mobile wireless networks. Merro [33] has proposed an untimed process calculus for mobile ad-hoc networks with a labelled characterisation of reduction barbed congruence, while [17] contains a calculus called CMAN, also with mobile ad-hoc networks in mind. This latter paper also gives a characterisation of reduction barbed congruence, this time in terms of a contextual bisimulation. It also contains a formalisation of an attack on the cryptographic routing protocol ARAN. Kouzapas and Philippou [27] have developed a theory of confluence for a calculus of dynamic networks and they use their machinery to verify a leader-election algorithm for mobile ad hoc networks.

Singh, Ramakrishnan and Smolka [48] have proposed the ω-calculus, a conservative extension of the π-calculus. A key feature of the ω-calculus is the separation of a node's communication and computational behaviour from the description of its physical transmission range. The authors provide a labelled transition semantics and a bisimulation in *open* style. The ω-calculus is then used for modelling the AODV ad-hoc routing protocol. Another extension of the π-calculus for modelling mobile wireless systems may be found in [7]; the calculus is used to verify reachability properties of the ad-hoc routing protocol LUNAR. Fehnker et al. [13] have proposed a process algebra for wireless mesh networks that combines novel treatments of local broadcast, conditional unicast and data structures. In this framework, they also model the AODV routing protocol and (dis)prove crucial properties such as loop freedom and packet delivery. Vigo et al. [53] have proposed a calculus of broadcasting processes that enables to reason about unsolicited messages and lacking of expected communication. Moreover, standard cryptographic mechanisms can be implemented in the calculus via term rewriting. The modelling framework is complemented by an executable specification of the semantics of the calculus in Maude.

All the calculi, mentioned up to now, except for [37], represent topological changes of mobile networks in the syntax. In contrast Ghassemi et al. [14] have proposed a process algebra called RBPT where topological changes to the connectivity graph are implicitly modelled in the operational semantics rather than in the syntax. They propose a notion of bisimulation for networks parametrised on a set of topological invariants that must be respected by equivalent networks. This work in then refined in [15] where the authors propose an equational theory for an extension of RBPT. Godskesen and Nanz [18] have proposed a simple timed calculus for wireless systems to express a wide range of mobility models.

A simple notion of time is also adopted in the calculus for wireless systems by Macedonio and Merro [31] to verify key management protocols for wireless sensor networks by applying semantics-based techniques. In [30] this notion of time is extended with probabilities. In this paper a probabilistic simulation theory is proposed to evaluate the performances gossip protocols in the context

of wireless sensor networks. Paper [50] also presents a probabilistic broadcast calculus for wireless networks where, unlike [30], nodes are mobile; due to mobility the connection probabilities may change. The authors examine the relation between a notion of weak bisimulation and a minor variant of PCTL*. Paper [10] investigate in detail the probabilistic behaviour of wireless networks. The paper presents a compositional theory based on a probabilistic generalisation of the well known may-testing and must-testing pre-orders. Also, it provides an extensional semantics to define both simulation and deadlock simulation preorders for wireless networks. Gallina et al. [8] propose a process algebraic model targeted at the analysis of both connectivity and communication interference in ad hoc networks. The framework includes a probabilistic process calculus and a suite of analytical techniques based on a probabilistic observational congruence and an interference-sensitive preorder. In particular, the preorder makes it possible to evaluate the interference level of different, behaviourally equivalent, networks. They use their framework to analyse the Alternating Bit Protocol. Song and Godskesen [51] introduce a continuous time stochastic broadcast calculus for mobile and wireless networks. The mobility between nodes in a network is modelled by a stochastic mobility function which allows to change part of a network topology depending on an exponentially distributed delay and a network topology constraint. They define a weak bisimulation congruence and apply their theory on a leader election protocol.

All the calculi mentioned up to now abstract away from the from the possibility of interference between broadcasts. Lanese and Sangiorgi [28] have instead proposed the CWS calculus, a lower level untimed calculus to describe interferences in wireless systems. In their operational semantics there is a separation between the beginning and ending of a broadcast, so there is some implicit representation of the passage of time. A more explicit timed generalisation of CWS is given [34] to express MAC-layer protocols such as CSMA/CA, where the authors propose a bisimilarity which is proved to be sound but not complete with respect to a notion of reduction barbed congruence. We view the current paper as a simplification and generalisation of [34].

The research we have mentioned so far has been focused on formalising various aspects of ad-hoc networks. However other than [18, 34], these various calculi abstract away from time. Nevertheless there is an extensive literature on timed process algebras, which we now briefly review. From a purely syntactic point of view, the earliest proposals are extensions of the three main process algebras, ACP, CSP and CCS. For example, [2] presents a real-time extension of ACP, [44] contains a denotational model for a timed extension of CSP, while CCS is the starting point for [36]. In [2] and [44] time is real-valued, and at least semantically, associated directly with actions. The other major approach to representing time is to introduce a special action to model the passage of time, and to assume that all other actions are instantaneous. This approach is advocated in [19, 5, 36, 39] and [55, 56], although the basis for this approach may be found in [6]. The current paper shares many of the assumptions of the languages presented in these papers; in particular we have been influenced by [22] which contains a timed version of CCS enjoying time determinism, maximal progress and patience. All the just mentioned papers assume that actions are instantaneous and only the extension of ACP presented in [19] does not incorporate time determinism; however maximal progress is less popular and patience is even rarer.

From this early work on timed process calculi a flourishing literature has emerged. Here we briefly mention some highlights of this research. Prasad [41] has proposed a timed variant of his CBS [40], called TCBS. In TCBS a timeout can force a process wishing to speak to remain idle for a specific interval of time; this corresponds to have a priority. TCBS also assumes time determinism and maximal progress. Corradini et al. [11] deal with *durational actions* proposing a framework relying on the notions of reduction and observability to naturally incorporate timing information in terms of process interaction. Our definition of timed reduction barbed congruence

takes inspiration from theirs. Corradini and Pistore [12] have studied durational actions to describe and reason about the performance of systems. Actions have lower and upper time bounds, specifying their possible different durations. Their *time equivalence* refines the untimed one. Baeten and Middelburg [3] consider a range timed process algebras within a common framework, related by embeddings and conservative extensions relations. These process algebras, ACP^{sat}, ACP^{srt}, ACP^{dat} and ACP^{drt}, allow the execution of two or more actions consecutively at the same point in time, separate the execution of actions from the passage of time, and consider actions to have no duration. The process algebra ACP^{sat} is a real-time process algebra with absolute time, ACP^{srt} is a real-time process algebra with relative time. Similarly, ACP^{dat} and ACP^{drt} are discrete-time process algebras with absolute time and relative time, respectively. In these process algebra the focus is on unsuccessful termination or deadlock. In [4] Baeten and Reniers extend the framework of [3] to model successful termination for the relative-time case. Laneve and Zavattaro [29] have proposed a timed extension of π-calculus where time proceeds asynchronously at the network level, while it is constrained by the local urgency at the process level. They propose a timed bisimilarity whose discriminating is weaker when local urgency is dropped.

References

[1] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci. Wireless sensor networks: a survey. *Computer Networks (Amsterdam, Netherlands: 1999)*, 38(4):393–422, March 2002.

[2] J. Baeten and J. Bergstra. Real Time Process Algebra. *Formal Aspects of Computing*, 3(2):142–188, 1991.

[3] J. Baeten and C. Middelburg. *Process Algebra with Timing*. EATCS Series. Springer-Verlag, 2002.

[4] J. C. M. Baeten and M. A. Reniers. Timed Process Algebra (With a Focus on Explicit Termination and Relative-Timing). In *SFM*, volume 3185 of *Lecture Notes in Computer Science*, pages 59–97. Springer-Verlag, 2004.

[5] J.C.M. Baeten and J.A. Bergstra. Discrete time process algebra. *Formal Aspects of Computing*, 8(2):188–208, 1996.

[6] G. Berry and L. Cosserat. The ESTEREL Synchronous Programming Language and its Mathematical Semantics. Technical Report 842, INRIA, Sophia-Antipolis, 1988.

[7] Johannes Borgström, Shuqin Huang, Magnus Johansson, Palle Raabjerg, Björn Victor, Johannes Åman Pohjola, and Joachim Parrow. Broadcast psi-calculi with an application to wireless protocols. In *SEFM*, volume 7041 of *Lecture Notes in Computer Science*, pages 74–89. Springer, 2011.

[8] Michele Bugliesi, Lucia Gallina, Andrea Marin, Sabina Rossi, and Sardaouna Hamadou. Interference-sensitive preorders for manets. In *Quantitative Evaluation of Systems (QEST), 2012 Ninth International Conference on*, pages 189–198. IEEE, 2012.

[9] Andrea Cerone. *Foundations of Ad Hoc Wireless Networks*. Ph.D Thesis, Trinity College Dublin, 2012.

[10] Andrea Cerone and Matthew Hennessy. Modelling probabilistic wireless networks. *Logical Methods in Computer Science*, 9(3), 2013.

[11] F. Corradini, G. Ferrari, and M. Pistore. On the semantics of durational actions. *Theoretical Computer Science*, 269(1-2):47–82, 2001.

[12] F. Corradini and M. Pistore. Closed interval process algebra versus interval process algebra. *Acta Informatica*, 37(7):467–509, 2001.

[13] A. Fehnker, R.J. van Glabbeek, P. Höfner, A. McIver, M. Portmann, and W. Lum Tan. A process algebra for wireless mesh networks. In *ESOP*, volume 7211 of *Lecture Notes in Computer Science*, pages 295–315. Springer, 2012.

[14] F. Ghassemi, W. Fokkink, and A. Movaghar. Restricted Broadcast Process Theory. In *SEFM*, pages 345–354. IEEE Computer Society, 2008.

[15] F. Ghassemi, W. Fokkink, and A. Movaghar. Equational Reasoning on Ad Hoc networks. In *FSEN*, volume 5961 of *Lecture Notes in Computer Science*, pages 113–128. Springer, 2009.

[16] F. Ghassemi, W. Fokkink, and A. Movaghar. Equational reasoning on mobile ad hoc networks. *Fundamenta Informaticae*, 105(4):375–415, 2010.

[17] J.C. Godskesen. A Calculus for Mobile Ad Hoc Networks. In *COORDINATION*, volume 4467 of *Lecture Notes in Computer Science*, pages 132–150. Springer Verlag, 2007.

[18] Jens Chr. Godskesen and Sebastian Nanz. Mobility Models and Behavioural Equivalence for Wireless Networks. In *COORDINATION*, volume 5521 of *Lecture Notes in Computer Science*, pages 106–122. Springer, 2009.

[19] J.F. Groote. Specification and Verification of Real Time Systems in acp. In *PSTV*, pages 261–274. North-Holland, 1990.
[20] Hennessy and Rathke. Bisimulations for a calculus of broadcasting systems. *TCS: Theoretical Computer Science*, 200(1–2):225–260, 1998.
[21] Matthew Hennessy. *A distributed Pi-calculus*. Cambridge University Press, 2007.
[22] Matthew Hennessy and Tim Regan. A process algebra for timed systems. *Information and Computation*, 117(2):221–239, March 1995.
[23] K. Honda and N. Yoshida. On reduction-based process semantics. *Theoretical Computer Science*, 152(2):437–486, 1995.
[24] IEEE 802.11 WG. ANSI/IEEE standard 802.11: Wireless LAN medium access control (MAC) and physical layer (PHY) specifications. IEEE Computer Society, 2007.
[25] A. Jeffrey and J. Rathke. Contextual equivalence for higher-order pi-calculus revisited. *Logical Methods in Computer Science*, 1(1:4), 2005.
[26] Raja Jurdak, Cristina Videira Lopes, and Pierre Baldi. A survey, classification and comparative analysis of medium access control protocols for ad hoc networks. *IEEE Communications Surveys and Tutorials*, 6(1-4):2–16, 2004.
[27] Dimitrios Kouzapas and Anna Philippou. A process calculus for dynamic networks. In *FMOODS/FORTE*, volume 6722 of *Lecture Notes in Computer Science*, pages 213–227. Springer, 2011.
[28] Ivan Lanese and Davide Sangiorgi. An operational semantics for a calculus for wireless systems. *Theor. Comput. Sci*, 411(19):1928–1948, 2010.
[29] C. Laneve and G. Zavattaro. Foundations of web transactions. In *FoSSaCS*, volume 3441 of *Lecture Notes in Computer Science*, pages 282–298. Springer, 2005.
[30] R. Lanotte and M. Merro. Semantic analysis of gossip protocols for wireless sensor networks. In J. Katoen and B. König, editors, *CONCUR*, volume 6901 of *Lecture Notes in Computer Science*, pages 156–170. Springer, 2011.
[31] D. Macedonio and M. Merro. A semantic analysis of wireless network security protocols. In A. Goodloe and S. Person, editors, *NASA Formal Methods*, volume 7226 of *Lecture Notes in Computer Science*, pages 403–417. Springer, 2012.
[32] Damiano Macedonio and Massimo Merro. A semantic analysis of wireless network security protocols. In Alwyn Goodloe and Suzette Person, editors, *NASA Formal Methods*, volume 7226 of *Lecture Notes in Computer Science*, pages 403–417. Springer, 2012.
[33] M. Merro. An Observational Theory for Mobile Ad Hoc Networks (full paper). *Information and Computation*, 207(2):194–208, 2009.
[34] Massimo Merro, Francesco Ballardin, and Eleonora Sibilio. A timed calculus for wireless systems. *Theor. Comput. Sci.*, 412(47):6585–6611, 2011.
[35] Milner and Sangiorgi. Barbed bisimulation. In *ICALP: Annual International Colloquium on Automata, Languages and Programming*, 1992.
[36] F. Moller and C. Tofts. A Temporal Calculus of Communicating Systems. In *CONCUR*, volume 458 of *Lecture Notes in Computer Science*, pages 401–415. Springer Verlag, 1990.
[37] S. Nanz and C. Hankin. A Framework for Security Analysis of Mobile Wireless Networks. *Theoretical Computer Science*, 367(1-2):203–227, 2006.
[38] Sebastian Nanz and Chris Hankin. Static analysis of routing protocols for ad-hoc networks, March 25 2004.
[39] Xavier Nicollin and Joseph Sifakis. The algebra of timed processes, atp: Theory and application. *Inf. Comput.*, 114(1):131–178, 1994.
[40] K. V. S. Prasad. A calculus of broadcasting systems. *Science of Computer Programming*, 25(2–3):285–327, December 1995. ESOP '94 (Edinburgh, 1994).
[41] K.V.S. Prasad. Broadcasting in Time. In *COORDINATION*, volume 1061 of *Lecture Notes in Computer Science*, pages 321–338. Springer Verlag, 1996.
[42] Theodore S. Rappaport. *Wireless communications - principles and practice*. Prentice Hall, 1996.
[43] Julian Rathke and Pawel Sobocinski. Deconstructing behavioural theories of mobility. In *Proc. Fifth IFIP International Conference On Theoretical Computer Science (TCS)*, volume 273 of *IFIP*, pages 507–520. Springer, 2008.
[44] G.M. Reed. A Hierarchy of Domains for Real-Time Distributed Computing. Technical Report, Oxford, 1988.
[45] D. Sangiorgi, N. Kobayashi, and E. Sumii. Environmental bisimulations for higher-order languages. In *Proceedings of the 22nd IEEE Symposium on Logic in Computer Science*, pages 293–302. IEEE Computer Society, 2007.
[46] D. Sangiorgi and D. Walker. *The π-calculus: a Theory of Mobile Processes*. Cambridge University Press, 2001.
[47] Davide Sangiorgi and David Walker. *The Pi-Calculus — A Theory of Mobile Processes*. Cambridge University Press, 2001.

[48] A. Singh, C.R. Ramakrishnan, and S.A. Smolka. A Process Calculus for Mobile Ad Hoc Networks. In *COORDINATION*, volume 5052 of *Lecture Notes in Computer Science*, pages 296–314, 2008.
[49] A. Singh, C.R. Ramakrishnan, and S.A. Smolka. A process calculus for mobile ad hoc networks. *SCP*, 75(6):440 – 469, 2010.
[50] L. Song and J.C. Godskesen. Probabilistic mobility models for mobile and wireless networks. In C.S. Calude and V/ Sassone, editors, *IFIP TCS*, volume 323 of *IFIP Advances in Information and Communication Technology*, pages 86–100. Springer, 2010.
[51] L. Song and J.C. Godskesen. Broadcast abstraction in a stochastic calculus for mobile networks. In J.C.M. Baeten, T. Ball, and F.S. de Boer, editors, *IFIP TCS*, volume 7604 of *Lecture Notes in Computer Science*, pages 342–356. Springer, 2012.
[52] Andrew S. Tanenbaum. *Computer Networks, 4th ed.* Prentice-Hall International, Inc., 2003.
[53] R. Vigo, F. Nielson, and H.R. Nielson. Broadcast, denial-of-service, and secure communication. In E.B. Johnsen and L. Petre, editors, *IFM*, volume 7940 of *Lecture Notes in Computer Science*, pages 412–427. Springer, 2013.
[54] Mengying Wang and Yang Lu. A timed calculus for mobile ad hoc networks. *arXiv preprint arXiv:1301.0045*, 2013.
[55] W. Yi. Real-Time Behaviour of Asynchronous Agents. In *CONCUR*, volume 458 of *Lecture Notes in Computer Science*, pages 502–520. Springer Verlag, 1990.
[56] W. Yi. *A Calculus of Real Time Systems*. Ph.D Thesis, Chalmers University, 1991.

Appendix A. Technical Definitions and Proofs of the Propositions

Definition A.1. [Process Environments] A process environment, is a mapping from process variables to system terms. In the following we use ρ to range over process environments. Given an open system term W and a process environment ρ, the (possibly open) system term $W\rho$ correspond to the system term obtained from W by replacing each free occurrence of any process variable X with $\rho(X)$. □

Lemma A.2. Let Γ be a channel environment, and W be an (open) system term whose free occurrences of process variables are time guarded. Then, given a channel c and two process environments ρ, ρ' such that both $(W\rho)$ and $(W\rho')$ are closed, $\mathsf{rcv}(\Gamma \triangleright W\rho, c) = \mathsf{rcv}(\Gamma \triangleright W\rho', c)$.

Proof. Note that if $\Gamma \vdash c : \mathbf{exp}$ then, for any channel environment ρ such that $W\rho$ is closed, we have that $\mathsf{rcv}(\Gamma \triangleright (W\rho), c) = \text{false}$, and there is nothing else left to prove.

Suppose then that $\Gamma \vdash c : \mathbf{idle}$, and let ρ, ρ' be two process environments such that $W\rho$ and $W\rho'$ are closed. We proceed by induction on the structure of W.

- $W = \lfloor c?(x).P \rfloor Q$. In this case we have $\mathsf{rcv}(\Gamma \triangleright (\lfloor c?(x).P \rfloor Q)\rho, c) = \mathsf{rcv}(\Gamma \triangleright (\lfloor c?(x).P \rfloor Q)\rho', c) = \text{true}$,
- $W = X$. This case is vacuous, as it contains an unguarded free occurrence of a process variable.
- $W = c\,!\langle e \rangle.P$. In this case $\mathsf{rcv}(\Gamma \triangleright (c\,!\langle e \rangle.P)\rho, c) = \mathsf{rcv}(\Gamma \triangleright c\,!\langle e \rangle.(P\rho), c) = \text{false}$, and $\mathsf{rcv}(\Gamma \triangleright (c\,!\langle e \rangle.P)\rho', c) = \mathsf{rcv}(\Gamma \triangleright c\,!\langle e \rangle.(P\rho'), c) = \text{false}$,
- $W = \tau.P$, $W = \sigma.P$, $W = [b]P, Q$, $W = \mathsf{nil}$ or $W = d[x].P$ where d is an arbitrary (possibly equal to c) channel; this case is analogous to the previous one,
- $W = P + Q$.Then we have that

$$\begin{aligned} \mathsf{rcv}(\Gamma \triangleright (P+Q)\rho, c) &= \mathsf{rcv}(\Gamma \triangleright (P\rho), c) \vee \mathsf{rcv}(\Gamma \triangleright (Q\rho), c) \\ &= \mathsf{rcv}(\Gamma \triangleright (P\rho'), c) \vee \mathsf{rcv}(\Gamma \triangleright (Q\rho'), c) \\ &= \mathsf{rcv}(\Gamma \triangleright (P+Q)\rho', c) \end{aligned}$$

where the equalities $\mathsf{rcv}(\Gamma \triangleright (P\rho), c) = \mathsf{rcv}(\Gamma \triangleright (P\rho'), c)$ and $\mathsf{rcv}(\Gamma \triangleright (Q\rho), c) = \mathsf{rcv}(\Gamma \triangleright (Q\rho'), c)$ follow by induction.

- $W = \mathsf{fix}\, X.P$. Then we have that

$$\begin{aligned} \mathsf{rcv}(\Gamma \triangleright (\mathsf{fix}\, X.P)\rho, c) &= \mathsf{rcv}(\Gamma \triangleright (P\rho), c) \\ &= \mathsf{rcv}(\Gamma \triangleright (P\rho'), c) \\ &= \mathsf{rcv}(\Gamma \triangleright (\mathsf{fix}\, X.P)\rho', c) \end{aligned}$$

  Again, the equality $\mathsf{rcv}(\Gamma \triangleright (P\rho), c) = \mathsf{rcv}(\Gamma \triangleright (P\rho'), c)$ follows by induction.
- $W = W_1 \mid W_2$. This case is analogous to the case $W = P + Q$,
- $W = \nu c : (t, \cdot).W'$. In this case $\mathsf{rcv}(\Gamma \triangleright (\nu c : (t, \cdot).W')\rho, c) = \mathsf{rcv}(\Gamma \triangleright (\nu c : (t, \cdot).W')\rho', c) = \mathsf{false}$,
- $W = \nu d : (t, \cdot).W'$, where $d \neq c$. Then we have

$$\begin{aligned} \mathsf{rcv}(\Gamma \triangleright (\nu d : (t, \cdot).W')\rho, c) &= \mathsf{rcv}(\Gamma \triangleright (W'\rho), c) \\ &= \mathsf{rcv}(\Gamma \triangleright (W'\rho'), c) \\ &= \mathsf{rcv}(\Gamma \triangleright (\nu d : (t, \cdot).W')\rho', c) \end{aligned}$$

□

Lemma A.3. Let Γ be a channel environment and W be an open system term where every free occurrence of process variables is guarded. Let also c be a channel and v be a value. There exists an open system term W' such that, for any process environment ρ for which $(W\rho)$ is closed, then $W'\rho$ is also closed, and $\Gamma \triangleright W\rho \xrightarrow{c?v} W'\rho$.

Proof. Note that if $\mathsf{rcv}(\Gamma \triangleright (W\rho), c = \mathsf{false})$ for some environment ρ, it suffices to choose $W' = W$. In fact, by Lemma A.2 we have that $\mathsf{rcv}(\Gamma \triangleright W\rho', c) = \mathsf{false}$ for any environment ρ' such that $W\rho'$ is closed. By applying Rule (RcvIgn) we obtain the transition $\Gamma \triangleright (W\rho') \xrightarrow{c?v} (W\rho')$.

Therefore, suppose that W is such that $\mathsf{rcv}(\Gamma \triangleright (W\rho), c) = \mathsf{true}$ for some process environment ρ (and, as a consequence of Lemma A.2, $\mathsf{rcv}(\Gamma \triangleright (W\rho'), c) = \mathsf{true}$ for any other process environment ρ'). Note that in this case we have that $\Gamma \vdash c : \mathbf{idle}$, and W cannot take the form $!\langle c\rangle.eP$, $\tau.P$, $\sigma.P$, $[b]P, Q$, nil or $d[x].P$. We check the remaining cases, by performing an induction on W. In the following ρ is an arbitrary process environment.

- Suppose that $W = \lfloor c?(x).P \rfloor Q$ for some processes P, Q. In this case we let $W' = c[x].P$. By definition $(\lfloor c?(x).P \rfloor Q)\rho = \lfloor c?(x).(P\rho') \rfloor (Q\rho)$, where $\rho' = \rho[x \mapsto x]$; by applying Rule (Rcv) we obtain that $\Gamma \triangleright (\lfloor c?(x).(P\rho') \rfloor (Q\rho) \xrightarrow{c?v} c[x].(P\rho')$. note that the latter system term can be rewritten as $(c[x].P)\rho$; note in fact that the process environments ρ and ρ' differ only at the entry for variable x, which is bound in $c[x].P$. Therefore we have the transition $\Gamma \triangleright (\lfloor c?(x).P \rfloor Q)\rho \xrightarrow{c?v} (c[x].P)\rho$.
- Suppose that $W = P + Q$. Note that, in order to ensure that $\mathsf{rcv}(\Gamma \triangleright (P + Q)\rho, c) = \mathsf{true}$, it must be either $\mathsf{rcv}(\Gamma \triangleright (P\rho), c) = \mathsf{true}$ or $\mathsf{rcv}(\Gamma \triangleright (Q\rho), c) = \mathsf{true}$. We consider only the first case, as the second one can be handled similarly. If $\mathsf{rcv}(\Gamma \triangleright (P\rho), c) = \mathsf{true}$ then by inductive hypothesis we have that there exists a system term W' such that $\Gamma \triangleright (P\rho) \xrightarrow{c?v} (W'\rho)$. By Rule (SumRcv), we can derive the transition $\Gamma \triangleright (P\rho) + (Q\rho) \xrightarrow{c?v} W\rho$, which can be rewritten as $\Gamma \triangleright (P+Q)\rho \xrightarrow{c?v} W'\rho$. Note also that if $\mathsf{rcv}(\Gamma \triangleright (P\rho), c) = \mathsf{true}$, then $\mathsf{rcv}(\Gamma \triangleright (P\rho'), c) = \mathsf{true}$ for any other process environment ρ', as a consequence of Lemma A.2, so that the choice of W' is independent from the process environment.
- Suppose that $W = \mathsf{fix}\, X.P$. By inductive hypothesis, there exists a process W'' such that, for any process environment ρ', $\Gamma \triangleright P\rho' \xrightarrow{c?v} W''\rho'$. In particular, let $\rho' = \rho[X \mapsto (\mathsf{fix}\, X.P)\rho]$, where ρ is an arbitrary process environment. We obtain that $\Gamma \triangleright P\rho[X \mapsto$

$(\mathsf{fix}\, X.P)\rho] \xrightarrow{c?v} W''\rho[X \mapsto (\mathsf{fix}\, X.P)\rho]$. $\Gamma \triangleright P\rho[X \mapsto (\mathsf{fix}\, X.P)\rho] = (\{\mathsf{fix}\, X.P/X\})P\rho$, and $W''\rho[X \mapsto (\mathsf{fix}\, X.P)\rho] = (\{\mathsf{fix}\, X.P/X\}W'')\rho$. Let then $W' = \{\mathsf{fix}\, X.P/X\}W'$. It suffices to apply Rule (Rec) to obtain the transition $\Gamma \triangleright (\mathsf{fix}\, X.P)\rho \xrightarrow{c?v} W'\rho$.

- Suppose that $W = W_1 \mid W_2$. By inductive hypothesis there exist W_1', W_2' such that $\Gamma \triangleright (W_1\rho) \xrightarrow{c?v} W_1'\rho$, and $\Gamma \triangleright (W_2\rho) \xrightarrow{c?v} W_2'\rho$. In this case we let $W' = W_1' \mid W_2'$. In fact, by Rule (RcvPar) it follows that $\Gamma \triangleright (W_1\rho) \mid (W_2\rho) \xrightarrow{c?v} (W_1'\rho) \mid (W_2'\rho)$, or equivalently $\Gamma \triangleright (W_1 \mid W_2)\rho \xrightarrow{c?v} (W_1' \mid W_2')\rho$.
- Finally, suppose $W = \nu d : (n, v).W_1$, where $d \neq c$. By inductive hypothesis we have that $\Gamma[d \mapsto (n, v)] \triangleright (W_1\rho) \xrightarrow{c?v} W'\rho$ for some W'. Now it suffices to apply Rule (ResI) to obtain $\Gamma \triangleright (W\rho) \xrightarrow{c?v} (W'\rho)$.

□

Proof of Proposition 2.9. Let $\Gamma \triangleright W$ be a configuration. First note that W is a closed system term, hence $W\rho = W$ for any process environment ρ. Given an arbitrary channel c and an arbitrary value v, Lemma A.3 ensures that there exists a system term W' such that $\Gamma \triangleright W \xrightarrow{c?v} W'$.

It remains to show that whenever $\Gamma \triangleright W \xrightarrow{c?v} W'$ for some W', if $\mathsf{rcv}(\Gamma \triangleright W, c) = \text{true}$ then $W' \neq W$; conversely, if $\mathsf{rcv}(\Gamma \triangleright W, c) = \text{false}$ then $W' = W$. This last statement can be proved by performing an induction on the proof of the derivation $\Gamma \triangleright W \xrightarrow{c?v} W'$; the proof is relatively simple, and the details are left to the reader.

The case where $\mathsf{rcv}(\Gamma \triangleright W, c) = \text{true}$ is slightly more complicated. In practice, we define a function $\#\mathsf{Rcv}(\cdot, c)$ which maps any system term into its number of active receivers along channel c and we show that, whenever $\Gamma \triangleright W \xrightarrow{c?v} W'$, then $\#\mathsf{Rcv}(W') > \#\mathsf{Rcv}(W)$. As an immediate consequence, $W' \neq W$. Formally, the function $\#\mathsf{Rcv}(\cdot, c)$ is defined inductively on the structure of system terms, by letting for any process P and system terms W_1, W_2,

(a) $\#\mathsf{Rcv}(P, c) = 0$,
(b) $\#\mathsf{Rcv}(d[x].P, c) = 1$ if $d = c$, 0 otherwise,
(c) $\#\mathsf{Rcv}(\nu d.(W_1), c) = \#\mathsf{Rcv}(W_1, c)$, when $d \neq c$,
(d) $\#\mathsf{Rcv}((W_1 \mid W_2), c) = \#\mathsf{Rcv}(W_1, c) + \#\mathsf{Rcv}(W_2, c)$.

We proceed by induction on the proof of the derivation $\Gamma \triangleright W \xrightarrow{c?v} W'$.

- The last rule applied in the proof of $\Gamma \triangleright W \xrightarrow{c?v} W'$ is Rule (Rcv). It follows that $W = \lfloor c?(x).P \rfloor Q$ for some processes P, Q, hence $\#\mathsf{Rcv}(W, c) = 0$. Further, $W' = c[x].P$, which leads to $\#\mathsf{Rcv}(W', c) = 1$;
- the last Rule applied in the proof of $\Gamma \triangleright W \xrightarrow{c?v} W'$ is (SumRcv); Then $W = P + Q$ for some processes P, Q such that $\mathsf{rcv}(\Gamma \triangleright P, c) = \text{true}$, and $\Gamma \triangleright P \xrightarrow{c?v} W'$. By definition we have that $\#\mathsf{Rcv}(P + Q, c) = 0$; also, $\#\mathsf{Rcv}(P, c) = 0$, hence by inductive hypothesis $\#\mathsf{Rcv}(W', c) > 0$, as we wanted to prove; the symmetric case of Rule (SumRcv) is handled similarly.
- the last rule applied in the proof of $\Gamma \triangleright W \xrightarrow{c?v} W'$ is Rule (Rec); this case is analogous to the previous one,
- the last rule applied in the proof of $\Gamma \triangleright W \xrightarrow{c?v} W'$ is Rule (ResV); then $W = \nu d.(W_1)$ and $W' = \nu d.(W_1')$ for some $d \neq c$, W_1 and W_1' such that $\Gamma \triangleright [d \mapsto (\cdot, \cdot)]W_1 \xrightarrow{c?v} W_1'$. In this case

we have that $\#\mathsf{Rcv}(\nu d.(W_1), c) = \#\mathsf{Rcv}(W_1, c) > \#\mathsf{Rcv}(W_1', c) = \#\mathsf{Rcv}(\nu d.(W_1'), c)$, where the inequality $\#\mathsf{Rcv}(W_1, c) > \#\mathsf{Rcv}(W_1', c)$ follows from the inductive hypothesis,

- the last case to analyse is the one in which Rule (RcvPar) has been applied last in the proof of $\Gamma \triangleright W \xrightarrow{c?v} W'$. Then $W = W_1 \mid W_2$ for some W_1, W_2 such that $\Gamma \triangleright W_1 \xrightarrow{c?v} W_1'$ and $\Gamma \triangleright W_2 \xrightarrow{c?v} W_2'$. Further, since we are assuming that $\mathsf{rcv}(\Gamma \triangleright W_1 \mid W_2, c) = \text{true}$, then either $\mathsf{rcv}(\Gamma \triangleright W_1, c) = \text{true}$ or $\mathsf{rcv}(\Gamma \triangleright W_2, c) = \text{true}$. Without loss of generality, suppose that $\mathsf{rcv}(\Gamma \triangleright W_1, c) = \text{true}$. Note that in this case, if $\mathsf{rcv}(\Gamma \triangleright W_2, c) = \text{false}$ then we know that $W_2' = W_2$, hence $\#\mathsf{Rcv}(W_2', c) = \#\mathsf{Rcv}(W_2, c)$. Otherwise, by inductive hypothesis it follows that $\#\mathsf{Rcv}(W_2', c) > \#\mathsf{Rcv}(W_2, c)$. In any case, we obtain that $\#\mathsf{Rcv}(W_2', c) \geq \#\mathsf{Rcv}(W_2, c)$. Also, by inductive hypothesis we have that $\#\mathsf{Rcv}(W_1', c) > \#\mathsf{Rcv}(W_1, c)$. By these two statements, and the definition of $\#\mathsf{Rcv}(\cdot, c)$, it follows that $\#\mathsf{Rcv}(W_1 \mid W_2, c) = \#\mathsf{Rcv}(W_1, c) + \#\mathsf{Rcv}(W_2, c) > \#\mathsf{Rcv}(W_1', c) + \#\mathsf{Rcv}(W_2, c) = \#\mathsf{Rcv}(W_1' \mid W_2', c)$.

□

Lemma A.4. Suppose that $\Gamma \triangleright W \xrightarrow{\sigma} W'$;

(i) if $W = P + Q$ for some processes P, Q then there exists two processes P', Q' such that $\Gamma \triangleright P \xrightarrow{\sigma} P'$, $\Gamma \triangleright Q \xrightarrow{\sigma} Q'$ and $W' = P' + Q'$,

(ii) if $W = W_1 \mid W_2$ for some W_1, W_2, then there exists two system terms W_1', W_2' such that $W' = W_1' \mid W_2'$, $\Gamma \triangleright W_1 \xrightarrow{\sigma} W_1'$ and $\gamma \triangleright W_2 \xrightarrow{\sigma} W_2'$.

Proof. Both statements can be proved by induction on the structure of W. We only provide the details for (i), since the proof for (ii) is identical in style.

- First note that if W is a basic process, that is it has either the form $\mathsf{nil}, c\,!\langle e\rangle.P, [b]P, Q$, $\lfloor c?(x).P \rfloor Q, \tau.P, \mathsf{fix}\, X.P$ or $\sigma.P$ then there is nothing to prove, as the assumption that $W = P + Q$ for some processes P, Q is not valid;
- suppose then that $W = P + Q$ for some processes P, Q, and that $\Gamma \triangleright P + Q \xrightarrow{\sigma} W'$. By inspecting the rules of the intensional semantics, it is clear that the last Rule applied in a proof of the transition above is (SumTime). Thus, there exist processes P_1, Q_1, P_1', Q_1' such that $P + Q = P_1 + Q_1$, $W' = P_1' + Q_1'$, $\Gamma \triangleright P_1 \xrightarrow{\sigma} P_1'$ and $\Gamma \triangleright Q_1 \xrightarrow{\sigma} Q_1'$. We need to show that there exist two processes P', Q' such that $\Gamma \triangleright P \xrightarrow{\sigma} P'$, $\Gamma \triangleright Q \xrightarrow{\sigma} Q'$ and $P' + Q' = P_1' + Q_1'$. Note that the assumption $P + Q = P_1 + Q_1$ leads to three possible cases:
  (1) there exists a process R such that $P_1 = P + R$, $Q = R + Q_1$; In this case we can apply the inductive hypothesis to the system term P_1 (note that P_1 is a smaller term than $P + Q$, as $P + Q = P_1 + Q_1$). Thus the transition $\Gamma \triangleright P_1 \xrightarrow{\sigma} P_1'$ ensures that there exist two system term P', R' such that $\Gamma \triangleright P \xrightarrow{\sigma} P'$, $\Gamma \triangleright R \xrightarrow{\sigma} R'$ and $P_1' = P' + R'$. Further, by applying Rule (SumTime) to the transitions $\Gamma \triangleright R \xrightarrow{\sigma} R'$ and $\Gamma \triangleright Q_1 \xrightarrow{\sigma} Q_1'$, we obtain $\Gamma \triangleright R + Q_1 \xrightarrow{\sigma} \Gamma \triangleright R' + Q_1'$. By letting $Q' = R' + Q_1'$, we can rewrite this last transition as $\Gamma \triangleright Q \xrightarrow{\sigma} Q'$. Finally notice that we have $W = P_1' + Q_1' = (P' + R') + Q_1' = P' + (R' + Q_1') = P' + Q'$, as we wanted to prove,
  (2) otherwise $P = P_1$ and $Q = Q_1$; this case is trivial, as it suffices to choose $P' = P_1', Q' = Q_1'$,
  (3) the last case possible is that there exists a process R such that $P = P_1 + R$, $Q_1 = R + Q$; the proof here is symmetrical to the first case, as now it is necessary to apply the inductive hypothesis to Q_1, rather than to P_1,

- the last remaining cases are those in which either $W = \nu c.W_1$ or $W = W_1 \mid W_2$. Again, these cases invalidate the hypothesis that W is a non-deterministic choice of processes, hence there is nothing to prove.

□

Proof of Proposition 2.10. We proceed by induction on the proof of the derivation $C \xrightarrow{\sigma} W_1$.

- The last rule applied in the derivation $C \xrightarrow{\sigma} W_1$ is rule (EndRcv). Then $C = \Gamma \triangleright c[x].P$ for some channel c, process P, channel environment Γ for which $\Gamma \vdash_{\mathrm{t}} c : 1$ and $\Gamma \vdash_{\mathrm{v}} c : w$ for some closed value w. Also $W_1 = \{w/x\}P$. Suppose now that $C \xrightarrow{\sigma} W_2$ for some system term W_2. By inspecting the rules of the intensional semantics we have that the only rule which could have been applied to infer this transition is again Rule (EndRcv). It follows that $W_2 = W_1 = \{w/x\}P$,
- the cases where the last rule applied in the proof of $C \xrightarrow{\sigma} W_1$ is either (TimeNil), (Sleep), (ActRcv) or (Timeout) can be proved similarly to the previous one,
- if the last rule applied in the proof of $C \xrightarrow{\sigma} W_1$ is (SumTime), then $C = \Gamma \triangleright P + Q$ for some processes P, Q. By Lemma A.4(i) we also know that $W_1 = P_1 + Q_1$ for some P_1, Q_1 such that $\Gamma \triangleright P \xrightarrow{\sigma} P_1, \Gamma \triangleright Q \xrightarrow{\sigma} Q_1$.

  Suppose that $C \xrightarrow{\sigma} W_2$ for some W_2. Then again, Lemma A.4(i) leads to $W_2 = P_2 + Q_2$ for some P_2, Q_2 such that $\Gamma \triangleright P \xrightarrow{\sigma} P_2$ and $\Gamma \triangleright Q \xrightarrow{\sigma} Q_2$. But by the inductive hypothesis we have that $P_1 = P_2$, $Q_1 = Q_2$. Hence $W_2 = P_2 + Q_2 = P_1 + Q_1 = W_1$,
- if Rule (Rec) has been applied last, then $W = \mathsf{fix}\, X.P$ for some process variable X and process P; further, $\Gamma \triangleright \{\mathsf{fix}\, X.P/X\}P \xrightarrow{\lambda} W_1$. Suppose now hat $\Gamma \triangleright \mathsf{fix}\, X.P \xrightarrow{\sigma} W_2$ for some W_2; then again the last rule applied has been (Rec), so that $conf\Gamma\{\mathsf{fix}\, X.P/X\}P \xrightarrow{\lambda} W_2$. Now, by the inductive hypothesis, we get that $W_1 = W_2$,
- the case where (ResV) is the last one in the derivation $C \xrightarrow{\sigma} W_1$ is similar in style to the previous one, and is therefore left to the reader,
- the last case is the one in which the last rule applied for deriving $C \xrightarrow{\sigma} W_1$ is Rule (TimePar); the proof in this case is analogous to the one where $C = \Gamma \triangleright P + Q$, using Lemma A.4(ii) instead of A.4(i).

□

Proof of Proposition 2.11. By induction on the proof of the transition. We only supply the details for the most interesting cases.

- The last Rule applied in the proof of the derivation $C \xrightarrow{\sigma} W_1$ is Rule (TimeOut). It follows that $C = \Gamma \triangleright \lfloor c?(x).P \rfloor Q$ for some Γ, channel c and processes P, Q such that $\Gamma \vdash c : \mathbf{idle}$. By inspecting the rules of the intensional semantics we note that no Rule can be applied to obtain a transition of the form $C \xrightarrow{c!v} W_2$, nor a transition of the form $C \xrightarrow{\tau} W_2$; for this last case, note in fact that a τ-action can be inferred for a configuration of the form $\Gamma \triangleright \lfloor c?(x).P \rfloor Q$ only via Rule (RcvLate), which however requires $\Gamma \vdash c : \mathbf{exp}$. This is in contrast with our assumption that $\Gamma \vdash c : \mathbf{idle}$.
- The last Rule applied in the proof of the transition $C \xrightarrow{\sigma} W_1$ is Rule (SumTime). Then $C = \Gamma \triangleright P + Q$ for some P, Q such that $\Gamma \triangleright P \xrightarrow{\sigma} P', \Gamma \triangleright Q \xrightarrow{\sigma} Q'$ and $W_1 = P' + Q'$.

We show, by contradiction, that $\Gamma \triangleright P + Q \xrightarrow{c!v}\!\!\!\!\!\!\rightarrow$ for any channel c and value v, and $\Gamma \triangleright P + Q \xrightarrow{\tau}\!\!\!\!\!\!\rightarrow$. So suppose that $\Gamma \triangleright P + Q \xrightarrow{\lambda} W_2$ for some system term W_2 and action $\lambda \in \{\tau, c!v \mid c \in \mathbf{Ch}, v \text{ closed value}\}$. Then the last rule applied in the proof of such a transition is either Rule (Sum) or its symmetric counterpart. In the first case we have that $\Gamma \triangleright P \xrightarrow{\lambda} W_2$, but this contradicts the inductive hypothesis; $\Gamma \triangleright P \xrightarrow{\sigma} P'$ implies $\Gamma \triangleright P \xrightarrow{c!v}\!\!\!\!\!\!\rightarrow$. Similarly, in the second case $\Gamma \triangleright Q \xrightarrow{\lambda} W_2$, which contradicts the inductive hypothesis applied to the transition $\Gamma \triangleright Q \xrightarrow{\sigma} Q'$. Therefore $\Gamma \triangleright P + Q \xrightarrow{\lambda}\!\!\!\!\!\!\rightarrow$.

□

Proof of Proposition 2.12. The proof is performed by induction on the structure of the proof of the derivation $\Gamma_1 \triangleright W \xrightarrow{\lambda} W'$. Again, we only consider the most interesting cases:

- The last rule applied in the proof of the derivation $\Gamma_1 \triangleright W \xrightarrow{\lambda} W'$ is Rule (Rcv). Then $\lambda = c?v$ for some channel c and value v, $\Gamma_1 \vdash c : \mathbf{idle}$, $W = \lfloor c?(x).P \rfloor Q$ for some P, Q and $W' = c[x].P$. By Hypothesis we have that $\Gamma_2 \vdash c : \mathbf{idle}$, so that $\Gamma_2 \triangleright \lfloor c?(x).P \rfloor Q \xrightarrow{c?v} c[x].P$.
- The last Rule applied in the proof of $\Gamma_1 \triangleright W \xrightarrow{\lambda} W'$ is Rule (RcvLate). Then $\lambda = \tau$, $\Gamma_1 \vdash c : \mathbf{exp}$ for some channel c, $W = \lfloor c?(x).P \rfloor Q$ and $W' = c[x].\{\mathsf{err}/x\}P$. By hypothesis $\Gamma_2 \vdash c : \mathbf{exp}$, so that Rule (RcvLate) can be applied leading to $\Gamma_2 \triangleright \lfloor c?(x).P \rfloor Q \xrightarrow{\lambda} c[x].\{\mathsf{err}/x\}P$.
- The last rule applied in the proof of $\Gamma_1 \triangleright W \xrightarrow{\lambda} W'$ is Rule (Then). Then $W = [b]P, Q$ for some b such that $[\![b]\!]_{\Gamma_1} = \text{true}$, $\lambda = \tau$ and $W' = \sigma.P$. Here it is necessary to make a case analysis on the form of the boolean expression b; the most interesting case, and the only one which we analyse, is $b = \exp(c)$ for some channel c. Since $[\![b]\!]_{\Gamma_1} = \text{true}$ then $\Gamma_1 \vdash c : \mathbf{exp}$. By hypothesis it follows that $\Gamma_2 \vdash c : \mathbf{exp}$, therefore $[\![b]\!]_{\Gamma_2} = \text{true}$. Now we can apply Rule (Then) to infer $\Gamma_2 \triangleright [b]P, Q \xrightarrow{\tau} \sigma.P$.
- The last rule applied in the proof of $\Gamma_1 \triangleright W \xrightarrow{\lambda} W'$ is Rule (Sync). It follows that $\lambda = c!v$ for some channel c and value v, $W = W_1 \mid W_2$ and $W' = W_1' \mid W_2'$ for some W_1, W_2, W_1', W_2' such that $\Gamma_1 \triangleright W_1 \xrightarrow{c!v} W_1'$, $\Gamma_2 \triangleright W_2 \xrightarrow{c!v} W_2'$. Then by inductive hypothesis we have that $\Gamma_2 \triangleright W_1 \xrightarrow{c!v} W_1'$ and $\Gamma_2 \triangleright W_2 \xrightarrow{c?v} W_2'$. An application of Rule (Sync) gives $\Gamma_2 \triangleright W \xrightarrow{c!v} W'$.

□

Proof of Proposition 2.13 (3)**.** Note that the proof of this statement uses Lemma, 2.13(2), which can be proved independently. For the if implication,suppose that $\Gamma \triangleright W_1 \xrightarrow{c!v} W_1'$ and $\Gamma \triangleright W_2 \xrightarrow{c?v} W_2'$. Then, by an application of Rule (Sync) we obtain that $\Gamma \triangleright W_1 \mid W_2 \xrightarrow{c!v} W_1' \mid W_2'$. Similarly, if $\Gamma \triangleright W_1 \xrightarrow{c?v} W_1'$ and $W_2 \xrightarrow{c!v} W_2'$, we can obtain the transition $\Gamma \triangleright W_1 \mid W_2 \xrightarrow{c!v} W_1' \mid W_2'$ using the symmetric counterpart of Rule (Sync).

For the only if implication, suppose that $\Gamma \triangleright W_1 \mid W_2 \xrightarrow{c!v} W'$. Note that we can rewrite $W_1 \mid W_2$ as $\prod_{i=1}^{k} P_k$ for some $k \geq 2$. We proceed by induction on k.

- $k = 2$. Then $W_1 = P_1$, $W_2 = P_2$. The last rule applied in the derivation of $\Gamma \triangleright P_1 \mid P_2 \xrightarrow{c!v} W'$ is either Rule (sync) or its symmetric counterpart. In the first case we obtain that $\Gamma \triangleright P_1 \xrightarrow{c!v} P_1'$, $\Gamma \triangleright P_2 \xrightarrow{c?v} P_2'$ and $W' = P_1' \mid P_2'$, so that there is nothing to prove. The second case is analogous.

- $k > 2$. Suppose that the statement is true for any index $i \leq k$. Again, the last rule applied in the proof of the transition $\Gamma \triangleright W_1 \mid W_2 \xrightarrow{c!v} W'$ is either Rule (Sync) or its symmetric counterpart. We consider only the first case, as the second one is treated similarly. If Rule (Sync) has been applied last, then there exist two system terms W_a, W_b such that $W_1 \mid W_2 = W_a \mid W_b$ and $\Gamma \triangleright W_a \xrightarrow{c!v} W'_a, \Gamma \triangleright W_b \xrightarrow{c!v} W'_b$ and $W' = W'_a \mid W'_b$. Since $W_a \mid W_b = W_1 \mid W_2$, we have three possible cases:
  - $W_1 = W_a \mid W_x$, $W_b = W_x \mid W_2$ for some system term W_x. Then we can apply Proposition 2.13 (2) to the transition $\Gamma \triangleright W_x \mid W_2 \xrightarrow{c?v} W'_b$ to show that $\Gamma \triangleright W_x \xrightarrow{c?v} W'_x$, $\Gamma \triangleright W_2 \xrightarrow{c?v} W'_2$ for some W'_x, W'_2 such that $W'_b = W'_x \mid W'_2$. Now we can apply Rule (Sync) to the transitions $\Gamma \triangleright W_a \xrightarrow{c!v} W'_a$ and $\Gamma \triangleright W_x \xrightarrow{c?v} W'_x$ to infer $\Gamma \triangleright W_1 \xrightarrow{c?v} W'_a \mid W'_x$. Let $W'_1 = W'_a \mid W'_x$. Then we have
$$W' = W'_a \mid W'_b = W'_a \mid W'_x \mid W'_2 = W'_1 \mid W'_2.$$
  - $W_a = W_1, W_b = W_2$. In this case there is nothing to prove, as it suffices to choose $W'_1 = W'_a, W'_2 = W'_b$ to obtain the result.
  - $W_a = W_1 \mid W_x$, $W_2 = W_x \mid W_b$ for some W_x. By the inductive hypothesis we obtain that either
    * $\Gamma \triangleright W_1 \xrightarrow{c!v} W'_1, \Gamma \triangleright W_x \xrightarrow{c?v} W'_x$ for some W'_1, W'_x such that $W'_a = W'_1 \mid W'_x$, or
    * $\Gamma \triangleright W_1 \xrightarrow{c?v} W'_1, \Gamma \triangleright W_x \xrightarrow{c!v} W'_x$ for some W'_1, W'_x such that $W'_a = W'_1 \mid W'_x$.

    We consider only the first case. In this case we can apply Rule (rcvPar) to the transitions $\Gamma \triangleright W_x \xrightarrow{c?v} W'_x$ and $\Gamma \triangleright W_b \xrightarrow{c?v} W'_b$ to obtain $\Gamma \triangleright W_2 \xrightarrow{c?v} W'_x \mid W'_b$. Let $W'_2 = W'_x \mid W'_b$. Then we have proved that $\Gamma \triangleright W_1 \xrightarrow{c!v} W'_1, \Gamma \triangleright W_2 \xrightarrow{c?v} W'_2$; further we have that
$$W' = W'_a \mid W'_b = W'_1 \mid W'_x \mid W'_b = W'_1 \mid W'_2$$
    as we wanted to prove. □

Proof of Lemma 4.1. We first prove that if $\Gamma \triangleright W \xrightarrow{\tau} \Gamma' \triangleright W'$ then $\Gamma \leq \Gamma'$. Note that such a transition could have been inferred in two different ways:

- via an application of Rule (TauExt), from which it follows that $\Gamma' = \mathrm{upd}_\tau(\Gamma) = \Gamma$, or
- via an application of Rule (Shh), applied to a transition of the form $\Gamma \triangleright W \xrightarrow{c!v} W'$; it follows that $\Gamma' = \mathrm{upd}_{c!v}(\Gamma)$, from which we obtain that $\Gamma \leq \Gamma'$.

Now suppose that $\Gamma \triangleright W \stackrel{\tau}{\Longrightarrow} \Gamma' \triangleright W'$. By definition, there exists an integer $n \geq 0$ such that $\Gamma \triangleright W = \Gamma_0 \triangleright W_0 \xrightarrow{\tau} \Gamma_1 \triangleright W_1 \xrightarrow{\tau} \cdots \xrightarrow{\tau} \Gamma_n \triangleright W_n = \Gamma' \triangleright W'$. By applying the result proved above to each step in this sequence, we obtain $\Gamma = \Gamma_0 \leq \Gamma_1 \leq \cdots \leq \Gamma_n = \Gamma'$, hence $\Gamma \leq \Gamma'$. □

Corollary A.5. For any channel c, $\Gamma \triangleright W \stackrel{\iota(c)}{\Longrightarrow}$ implies $\Gamma \triangleright W \xrightarrow{\iota(c)}$.

Proof. By Definition, $\Gamma \triangleright W \Longrightarrow \Gamma' \triangleright W' \xrightarrow{\iota(c)}$ for some Γ', W'. Since, $\Gamma' \triangleright W' \xrightarrow{\iota(c)}$ we obtain that $\Gamma' \vdash c : \mathbf{idle}$. Now Lemma 4.1 gives $\Gamma \leq \Gamma'$, hence $\Gamma \vdash c : \mathbf{idle}$. Therefore we can apply Rule (Idle) of the extensional semantics and derive $\Gamma \triangleright W \xrightarrow{\iota(c)} \Gamma \triangleright W$. □

Proof of Lemma 4.2. Suppose $\Gamma_1 \triangleright W_1 \approx \Gamma_2 \triangleright W_2$. If $\Gamma_1 \vdash c : \mathbf{idle}$ then by definition of Rule (Idle) of Table 6 it follows that $\Gamma_1 \triangleright W_1 \overset{\iota(c)}{\longmapsto}$. As $\Gamma_1 \triangleright W_1 \approx \Gamma_2 \triangleright W_2$, it follows that $\Gamma_2 \triangleright W_2 \overset{\iota(c)}{\Longmapsto}$. From Corollary A.5 we have that $\Gamma_2 \triangleright W_2 \overset{\iota(c)}{\longmapsto}$, and by the definition of Rule (Idle) that $\Gamma_2 \vdash c : \mathbf{idle}$. □

Proof of Lemma 4.7. We have to show that if C is well-formed and $C \xrightarrow{\lambda} W'$, then $C' = \mathrm{upd}_\lambda(\Gamma) \triangleright W'$ is also well-formed. We provide the details of the most interesting cases of a rule induction on the proof of the aforementioned transition.

- The last rule applied is Rule (Rcv). Then $\lambda = c?v$ for some channel c and closed value v. Further, $C = \Gamma \triangleright \lfloor c?(x).P \rfloor Q$, $W' = c[x].P$ and $\mathrm{upd}_{c?v}(\Gamma) \vdash c : \mathbf{exp}$. The second equation in Definition 4.6 ensures that $C' \in$ Wnets,
- the last rule applied is Rule (EndRcv); in this case $\lambda = \sigma$, $W = c[x].P$ for some c such that $\Gamma \vdash c : \mathbf{exp}$, and $W' = \{w/x\}P$, where w is the closed value such that $\Gamma \vdash_{\mathrm{v}} c : w$. It follows from the first equation in Definition 4.6 that $C' = \mathrm{upd}_\sigma(\Gamma) \triangleright W'$ is well formed,
- the last rule applied is Rule (ActRcv). In this case $W = W' = c[x].P$ for some c such that $\Gamma \vdash_{\mathrm{t}} c : n$, where $n > 1$. To show that $C' = \mathrm{upd}_\sigma(\Gamma) \triangleright c[x].P$, it suffices to prove that $\mathrm{upd}_\sigma(\Gamma) \vdash c : \mathbf{exp}$; but this is true, since by Definition of $\mathrm{upd}_\sigma(\cdot)$ we have that $\mathrm{upd}_\sigma(\Gamma) \vdash_{\mathrm{t}} c : n - 1$, and now $n - 1 > 0$,
- the last rule applied is Rule (Sync). Then $\lambda = c!v$, $W = W_1 \mid W_2$, $W' = W_1' \mid W_2'$ for some W_1, W_2, W_1', W_2' such that $\Gamma \triangleright W_1 \xrightarrow{c!v} W_1'$, $\Gamma \triangleright W_2 \xrightarrow{c?v} W_2'$ and $W' = W_1' \mid W_2'$. By inductive hypothesis the configurations $C_1 = \mathrm{upd}_{c!v}(\Gamma) \triangleright W_1'$ and $C_2 = \mathrm{upd}_{c?v}(\Gamma) \triangleright W_2'$ are well formed, so by the third equation in Definition 4.6 we have that $C' \in$ Wnets[6]. □

Proof of Proposition 4.8. Let $\Gamma \triangleright W$ be a well-formed configuration. We give the details of the most important cases of a structural induction performed on the structure of a system term W.

- $W = c\,!\langle v \rangle.P$, or $W = \tau.P$; this case is vacuous, since by definition of instantaneous reductions $\Gamma \triangleright W \not\rightarrow_i$,
- $W = \sigma.P$; this case is trivial, since by applying Rule (Sleep) we infer $\Gamma \triangleright W \xrightarrow{\sigma} P$, hence $\Gamma \triangleright W \rightarrow_\sigma \mathrm{upd}_\sigma(\Gamma) \triangleright P$,
- $W = c[x].P$. By definition of well-formed networks we have that $\Gamma \vdash c : \mathbf{exp}$. Then there are two possible cases:
  - $\Gamma \vdash_{\mathrm{t}} c : 1$ and $\Gamma \vdash_{\mathrm{v}} c : v$ for some value v. We can apply Rule (EndRcv) to infer the transition $\Gamma \triangleright c[x].P \xrightarrow{\sigma} \{v/x\}P$, which in turns gives the reduction $\Gamma \triangleright c[x].P \rightarrow_\sigma \mathrm{upd}_\sigma(\Gamma) \triangleright \{v/x\}P$,
  - $\Gamma \vdash_{\mathrm{t}} c : n$ for some $n > 1$; in this case we can apply Rule (ActRcv) to infer $\Gamma \triangleright c[x].P \xrightarrow{\sigma} c[x].P$, leading to $\Gamma \triangleright c[x].P \rightarrow_\sigma \mathrm{upd}_\sigma(\Gamma) \triangleright c[x].P$.
- $W = \mathsf{fix}\, X.P$. Recall that in this case every occurrence of the process variable X in P is (time) guarded, so that we can apply the inductive hypothesis to the term $\{\mathsf{fix}\, X.P/X\}P$. Now suppose that $\Gamma \triangleright \mathsf{fix}\, X.P \not\rightarrow_i$. Then it follows that $\Gamma \triangleright \{\mathsf{fix}\, X.P/X\}P \not\rightarrow_i$, and by inductive hypothesis $\Gamma \triangleright \{\mathsf{fix}\, X.P/X\}P \rightarrow_\sigma$. Now it is easy to show that $\Gamma \triangleright \mathsf{fix}\, X.P \rightarrow_\sigma$.
- $W = P + Q$. Suppose that $\Gamma \triangleright P + Q \not\rightarrow_i$. That is, $\Gamma \triangleright P \not\rightarrow_i$, $\Gamma \triangleright Q \not\rightarrow_i$, By inductive hypothesis we have that $\Gamma \triangleright P \xrightarrow{\sigma} P'$, $\Gamma \triangleright Q \xrightarrow{\sigma} Q'$ for some P', Q'. It follows from Rule (SumTime) that $\Gamma \triangleright P + Q \xrightarrow{\sigma} P' + Q'$, hence $\Gamma \triangleright P + Q \rightarrow_\sigma \mathrm{upd}_\sigma(\Gamma) \triangleright P' + Q'$. □

[6]Recall that $\mathrm{upd}_{c!v}(\Gamma) = \mathrm{upd}_{c?v}(\Gamma)$.

Proposition A.6. For any channel environment Γ, (possibly open) process P and process environment ρ such that $P\rho$ is closed, then $\Gamma \triangleright P\rho$ is well-timed.

Proof. We give the details of the most important cases of an induction performed on the structure of the process W. In the following we assume that ρ is a process environment such that $W\rho$ is closed; recall that we are assuming that free occurrences of process variables are time guarded in W.

- $W = \lfloor c?(x).P \rfloor Q$. Then we have that $\Gamma \triangleright (\lfloor c?(x).P \rfloor Q)\rho \not\mapsto_i$; it follows that $\Gamma \triangleright (\lfloor c?(x).P \rfloor Q)\rho$ is well-timed.
- $W = X$ for some process variable X; this case is vacuous, since it violates the assumption that free occurrences of process variables are (time) guarded in W,
- $W = \mathsf{fix}\, X.P$ for some process P. Let ρ' be the environment defined as $\rho[X \mapsto (\mathsf{fix}\, X.P)\rho]$. By inductive hypothesis we have that $\Gamma \triangleright P\rho'$ is well-timed. Further, by definition $P\rho' = (\{\mathsf{fix}\, X.P/X\}P)\rho$. Now note that $\Gamma \triangleright (\mathsf{fix}\, X.P)\rho \rightarrow^h C'$ if and only if $\Gamma \triangleright (\mathsf{fix}\, X.P/X\}P)\rho \rightarrow^h C'$. It follows that $\Gamma \triangleright (\mathsf{fix}\, X.P)\rho$ is well-timed.
- $W = P + Q$. Suppose that both $(P + Q)\rho$ is closed; that is, both $P\rho$ and $Q\rho$ are closed. By inductive hypothesis they are well timed, meaning that there exists $k_P \geq 0$ such that whenever $\Gamma \triangleright P\rho \rightarrow^h \Gamma' \triangleright P'$ then $h \leq k_P$; similarly, there exists $k_Q \geq 0$ such that whenever $\Gamma \triangleright Q\rho \rightarrow^h \Gamma' \triangleright Q'$ for some h, then $h \leq k_Q$. Choose $k = \max(k_P, k_Q)$. It is easy to show that whenever $\Gamma \triangleright (P + Q)\rho \rightarrow^h \Gamma' \triangleright W'$ then either $\Gamma \triangleright P\rho \rightarrow^h \Gamma' \triangleright W'$, in which case $h \leq k_P \leq k$, or $\Gamma \triangleright Q\rho \rightarrow^h \Gamma' \triangleright W'$, in which case $h \leq k_P \leq k$. It follows that $\Gamma \triangleright (P + Q)\rho$ is well-timed.

□

Proof of Proposition 4.12. We give the proof for a fragment of the language where channel restriction is omitted. This limitation is needed only to avoid technical complications in the proof of the statement. In fact, when channel restriction is present, we need to introduce a structural congruence $\equiv$ between system terms; the main property required by this relation is that it preserves transitions of configurations, meaning that whenever $W_1 \equiv W_2$ and $\Gamma \triangleright W_1 \xrightarrow{\lambda} W_1'$, then $\Gamma \triangleright W_2 \xrightarrow{\lambda} W_2'$, with $W_2 \equiv W_2'$. Also, the relation $\equiv$ needs to be defined so that any system term W can be rewritten in the form $\nu\tilde{c}.\left(\prod_{i=1}^n P_i\right)$. See [9], Definition **9.1.2** at Page 174, for the definition of the structural congruence .

Let us focus on the case in which channel restriction is not present in our language First note that the result holds for any well-formed configuration of the form $\Gamma \triangleright P$, where P is a closed process; in fact we have that, $\Gamma \triangleright P = \Gamma \triangleright P\rho$ for any process environment ρ, and the latter is well-timed by Proposition A.6.

Otherwise, we can rewrite $\Gamma \triangleright W$ as $\Gamma \triangleright \prod_{i=1}^n P_i$, for some processes $P_1, \cdots, P_n$. Note that each configuration $\Gamma_i \triangleright P_i$ is well-formed, hence well-timed; by definition there exists an index $k_{P_i} \geq 0$ such that, whenever $\Gamma \triangleright P_i \rightarrow_i^h \Gamma' \triangleright P_i'$, then $h \leq k_{P_i}$. Now suppose that $\Gamma \triangleright \prod_{i=1}^n P_i \rightarrow_i^h \Gamma' \triangleright \prod_{i=1}^n P_i'$; we show that $h \leq \left(\sum_{i=1}^n k_{P_i}\right)$ by induction on h.

The case $h = 0$ is trivial; suppose then that $h > 0$, and the statement is valid for $h-1$; in this case we can rewrite the (weak) reduction above as $\Gamma \triangleright \prod_{i=1}^n P_i \rightarrow_i \Gamma'' \triangleright \prod_{i=1}^n P_i'' \rightarrow_i^{h-1} \Gamma' \triangleright \prod_{i=1}^n P_i'$, and by inductive hypothesis $h - 1 \leq \sum_{i=1^n} k_{P_i''}$. Let us focus on why $\Gamma \triangleright \prod_{i=1}^n P_i \rightarrow_i \Gamma'' \triangleright \prod_{i=1}^n P_i'' \rightarrow_i^{h-1}$.

(i) $\Gamma \triangleright \prod_{i=1}^n P_i \xrightarrow{\tau} \prod_{i=1}^n P_i''$, and $\Gamma'' = \Gamma$; in this case it is not difficult to note that there exists an index $j : 1 \leq j \leq n$ such that $\Gamma \triangleright P_j \xrightarrow{\tau} P_j''$, and for any index $i \neq j, 1 \leq i \leq n$, $P_i'' = P_i$. In this case we have that $k_{P_j''} \leq k_{P_j} - 1$

Without loss of generality, let $j = 1$. Then we have that

$$\begin{aligned} h - 1 \leq & \sum_{i=1}^{n} k_{P_i''} & = \\ = & k_{P_1''} + & \sum_{i=2}^{n} k_{P_i} \leq \\ \leq & (k_{P_1} - 1) + & \sum_{i=2}^{n} k_{P_i} = \\ = & \left(\sum_{i=1}^{n} k_{P_i}\right) - 1 & \end{aligned}$$

Hence $h \leq \left(\sum_{i=1}^{n} k_{P_i}\right)$, as we wanted to prove;

(ii) Otherwise $\Gamma \triangleright \prod_{i=1}^{n} P_i \xrightarrow{c!v} \prod_{i=1}^{n} P_i''$, and $\Gamma'' = \mathrm{upd}_{c!v|}(\Gamma)$. In this case we can partition the set $\{1, \cdots, n\}$ into three sets $\{l\}$, I and J such that **(a)** $\Gamma \triangleright P_l \xrightarrow{c!v} \Gamma'' \triangleright P_l''$ and $P'' = \sigma^{\delta_v}.Q$ for some process Q, **(b)** for any $i \in I$, $\mathsf{rcv}(\Gamma \triangleright P_i, c) = \text{true}$ and $P_i'' = c[x].Q_i$ for some process Q_i, **(c)** for any $j \in J$, $\mathsf{rcv}(\Gamma \triangleright P_j, c) = \text{false}$ and $P_j'' = P_j$. Note that **(a)** implies that $k_{P_l''} = 0$ and $1 \leq k_{P_l}$, **(b)** implies that $k_{P_i''} = 0$ for any $i \in I$ and **(c)** implies that $k_{P_j''} = k_{P_j}$ for any $j \in J$.

Without loss of generality, suppose that $l = 1$, $I = \{2, \cdots, m\}$ for some $m \leq n$, and $J = \{m + 1, \cdot n\}$. In this case we have

$$\begin{aligned} h - 1 \leq & & \sum_{i=1}^{n} k_{P_i''} & & = \\ = & k_{P_1''} + & \left(\sum_{i=2}^{m} k_{P_i''}\right) + & \left(\sum_{i=m+1}^{n} k_{P_i''}\right) = \\ = & 0 + & 0 + & \sum_{i=m+1}^{n} k_{P_i} \leq \\ \leq & (k_{P_1} - 1) + & 0 + & \sum_{i=m+1}^{n} k_{P_i} \leq \\ \leq & & \sum_{i=1}^{n} k_{P_i} & & \end{aligned}$$

Again the last inequation gives $h \leq \left(\sum_{i=1}^{n} k_{P_i}\right)$. □

Lemma A.7. Let us say that a system term T is behaviourally independent from W if each channel name appearing free in T does not appear free in W, and vice versa.

If T is independent from a configuration W, then whenever $\Gamma \triangleright W \mid T \rightarrowtriangle_i C$, then either

(i) $C = \Gamma' \triangleright W \mid T'$, and $\Gamma \triangleright T \rightarrowtriangle_i \Gamma' \triangleright W'$, or
(ii) $C = \Gamma' \triangleright W' \mid T$, and $\Gamma \triangleright T \rightarrowtriangle_i \Gamma' \triangleright W'$.

Proof. Suppose that T is a system term independent from a configuration $\Gamma \triangleright W$, and that $\Gamma \triangleright W \mid T \rightarrowtriangle_i C$. By the definition of instantaneous reductions, there are two possibilities:

(1) $\Gamma \triangleright W \mid T \xrightarrow{\tau} \widehat{W}$, and $C = \Gamma \triangleright \widehat{W}$. By Proposition 2.13(1) then either $\widehat{W} = W' \mid T$, and $\Gamma \triangleright W \xrightarrow{\tau} W'$, or $\widehat{W} = W \mid T'$, and $\Gamma \triangleright T \xrightarrow{\tau} T'$; in the first case we obtain the reduction $\Gamma \triangleright W \mid T \rightarrow_i \Gamma \triangleright W' \mid T$, while in the second one we get $\Gamma \triangleright W \mid T \rightarrow_i \Gamma \triangleright W \mid T'$,

(2) the second possibility is that $\Gamma \triangleright W \mid T \xrightarrow{c!v} \widehat{W'}$, and $C = \Gamma' \triangleright \widehat{W}$, where $\Gamma' = \mathrm{upd}_{c!v}(\Gamma)$. In this case, by Proposition 2.133 then $\widehat{W} = W' \mid T'$ and either

   (a) $\Gamma \triangleright W \xrightarrow{c!v} W'$, $\Gamma \triangleright T \xrightarrow{c!v} T'$; the first transition is possible only if c appears free in W, which by assumption gives that c does not appear free in T; it follows that $\mathsf{rcv}(\Gamma \triangleright T, c) = \text{false}$, and by Lemma 2.9 we obtain that $T' = T$. By converting the intensional transition in a reduction (recalling that $\Gamma' = \mathrm{upd}_{c!v}(\Gamma)$), we obtain that $\Gamma \triangleright W \mid T \rightarrow_i \Gamma' \triangleright W' \mid T$,

   (b) or $\Gamma \triangleright W \xrightarrow{c?v} W'$, $\Gamma \triangleright T \xrightarrow{c!v} T'$; this case can be handled symmetrically to the previous one, and leads to $\Gamma \triangleright W \mid T \rightarrow_i \Gamma' \triangleright W \mid T'$.

□

Lemma A.8. Let $\Gamma_1 \triangleright W$ be a configuration, and let Γ_2 be a channel environment such that, for any channel c appearing free in W, $\Gamma_2(c) = \Gamma_1(c)$. Then if $\Gamma_1 \triangleright W \rightarrow \Gamma_1' \triangleright W'$, there exists a channel environment Γ_2' such that $\Gamma_2 \triangleright W \rightarrow \Gamma_2' \triangleright W_2'$, and $\Gamma_1'(c) = \Gamma_2'(c)$ for any c appearing free in W.

Outline of the proof. The reduction $\Gamma_1 \triangleright W \rightarrow_i \Gamma_1' \triangleright W'$ can be converted in a transition of the form $\Gamma_1 \triangleright W \xrightarrow{\lambda} W'$, where λ takes either the form τ, $c!v$ or σ. Note here that if λ takes the form $c!v$, then c appears free in W. By performing an induction on the proof of the derivation of this transition we can infer a transition for the configuration $\Gamma_2 \triangleright W$, namely $\Gamma_2 \triangleright W \xrightarrow{\lambda} W'$. Also, by letting $\Gamma_2' = \mathrm{upd}_\lambda(\Gamma_2)$, we obtain the reduction $\Gamma_2 \triangleright W \rightarrow \Gamma_2' \triangleright W'$. Now it remains to note that if c appears free then, by hypothesis, $\Gamma_1(c) = \Gamma_2(c)$; hence $\Gamma_1'(c) = \mathrm{upd}_\lambda(\Gamma_1)(c) = \mathrm{upd}_\lambda(\Gamma_2)(c) = \Gamma_2'(c)$. □

Corollary A.9. [Independence of Computations] Let $\Gamma \triangleright W$ be a configuration, and let T be a system term which only uses fresh channels. Then whenever $\Gamma \triangleright W \mid T \rightarrow^* \Gamma'' \triangleright \widehat{W}$ it follows that $\widehat{W} = W' \mid T'$ for some W', T' such that $\Gamma \triangleright W' \rightarrow^* \Gamma' \triangleright W'$, where Γ' is such that $\Gamma'(c) = \Gamma''(c)$ for any c appearing free in W.

Outline. By induction on the number of derivations k in a sequence of k reductions, $\Gamma \triangleright W \mid T \rightarrow^k \Gamma'' \triangleright \widehat{W}$; in the inductive step it is necessary to distinguish whether the first reduction of the sequence is instantaneous or timed. In the first case, the result follows from lemmas A.7 and A.8. In the second case, we need to recover the timed transitions for the individual components $\Gamma \triangleright W$ and $\Gamma \triangleright T$, then apply Lemma A.8. □

Proof of Lemma 4.14 (Outline). This is a variation on analogous results already given in the literature, for a number of different process calculi. We show that the relation

$$\begin{aligned}\mathcal{S} = \{&(\Gamma_1 \triangleright W_1, \Gamma_2 \triangleright W_2) \ : \\ &\Gamma_1' \triangleright W_1 \mid T_1 \simeq \Gamma_2' \triangleright W_2 \mid T_2 \text{ for some } T_1, T_2 \text{ independent from both } W_1, W_2 \\ &\text{and } \Gamma_1 \triangleright (c) = \Gamma_1'(c), \Gamma_2(c) = \Gamma_2'(c) \text{ whenever } c \text{ appears free in } W\}\end{aligned}$$

is barb preserving, reduction closed and contextual. Note that it is necessary to employ Corollary A.9 to prove that $\mathcal{S}$ is reduction closed. □

Proof of Proposition 4.15: The two statements are proved separately. Let $\Gamma_1 \triangleright W_1, \Gamma_2 \triangleright W_2$ be well-formed, and suppose that $\Gamma_1 \triangleright W_1 \simeq \Gamma_2 \triangleright W_2$.

(1) Suppose that $\Gamma_1 \triangleright W_1 \rightarrow\!\!\triangleright_i \Gamma_1' \triangleright W_1'$. We have two possible cases, according to the definition of $\rightarrow\!\!\triangleright_i$:

(i) $\Gamma_1 \triangleright W_1 \xrightarrow{\tau} W_1'$ and $\Gamma_1' = \mathrm{upd}_\tau(\Gamma_1) = \Gamma_1$, by an application of rule (TauExt)

(ii) $\Gamma_1 \triangleright W_1 \xrightarrow{c!v} W_1'$ and $\Gamma_1' = \mathrm{upd}_{c!v}(\Gamma_1)$, by an application of rule (Shh).

We consider the first case; the proof for the second case is virtually identical. Let *eureka* be a fresh channel; that is it does not appear free in W_1 and must satisfy $\Gamma_1 \vdash \mathit{eureka} : \mathbf{idle}$. Let *ok* be a message which requires one time unit to be transmitted, i.e. $\delta_{ok} = 1$. By an application of rules (TauPar) and (TauExt) we derive

$$\Gamma_1 \triangleright W_1 \mid \mathit{eureka}!\langle ok\rangle \stackrel{\tau}{\longmapsto} \Gamma_1' \triangleright W_1' \mid \mathit{eureka}!\langle ok\rangle$$

with $\Gamma_1' \triangleright W_1' \mid \mathit{eureka}!\langle ok\rangle \Downarrow_{\mathit{eureka}}$ and $\Gamma_1' \vdash \mathit{eureka} : \mathbf{idle}$. By Definition 2.14 this transition corresponds in the reduction semantics to

$$\Gamma_1 \triangleright W_1 \mid \mathit{eureka}!\langle ok\rangle \rightarrow\!\!\triangleright \Gamma_1' \triangleright W_1' \mid \mathit{eureka}!\langle ok\rangle$$

As $\Gamma_1 \triangleright W_1 \simeq \Gamma_2 \triangleright W_2$ and $\simeq$ is contextual, this step must be matched by a sequence of reductions

$$\Gamma_2 \triangleright W_2 \mid \mathit{eureka}!\langle ok\rangle \rightarrow\!\!\triangleright^* C \tag{A.1}$$

such that $\Gamma_1' \triangleright W_1' \mid \mathit{eureka}!\langle ok\rangle \simeq C$. Depending on whether the transmission at *eureka* is part of the sequence of reductions or not, the configuration C must be one of the following:

$$\begin{array}{lcll} C_1 & = & \Gamma_2' \triangleright W_2' \mid \mathit{eureka}!\langle ok\rangle & \text{with} \quad \Gamma_2' \vdash \mathit{eureka} : \mathbf{idle} \\ C_2 & = & \Gamma_2' \triangleright W_2' \mid \sigma.\mathsf{nil} & \text{with} \quad \Gamma_2' \vdash \mathit{eureka} : \mathbf{exp} \\ C_3 & = & \Gamma_2' \triangleright W_2' \mid \mathsf{nil} & \text{with} \quad \Gamma_2' \vdash \mathit{eureka} : \mathbf{idle} \end{array}$$

As *eureka* is a fresh channel (hence not appearing free in W_2, it follows that $C_3 \not\Downarrow_{\mathit{eureka}}$; therefore C cannot be C_3. Since $\Gamma_1' \triangleright W_1' \mid \mathit{eureka}!\langle ok\rangle \simeq C$ and $\Gamma_1' \vdash \mathit{eureka} : \mathbf{idle}$, by Proposition 4.13 (which can be applied since we are assuming that C is well-formed, hence well-timed) it follows that C cannot be C_2. So, the only possibility is $C = C_1$. By Lemma 4.14 it follows that $\Gamma_1' \triangleright W_1' \simeq \Gamma_2' \triangleright W_2'$. It remains to show that $\Gamma_2 \triangleright W_2 \rightarrow\!\!\triangleright_i^* \Gamma_2' \triangleright W_2'$.

To this end we can extract out from the reduction sequence (A.1) above a reduction sequence

$$\Gamma_2 \triangleright W_2 \rightarrow\!\!\triangleright^* \Gamma_2' \triangleright W_2'$$

We show that each step in this sequence, say $\Gamma \triangleright W \rightarrow\!\!\triangleright \Gamma' \triangleright W'$, corresponds to an instantaneous reduction, $\Gamma \triangleright W \rightarrow\!\!\triangleright_i \Gamma' \triangleright W'$, from which the result follows.

Recall from Definition 2.14 that there are three possible ways to infer the reduction step $\Gamma \triangleright W \rightarrow\!\!\triangleright \Gamma' \triangleright W'$. If it is either (Internal), i.e. $\Gamma \triangleright W \xrightarrow{\tau} W'$, or a (Transmission), i.e. $\Gamma \triangleright W \xrightarrow{c!v} W'$, then by definition $\Gamma \triangleright W \rightarrow\!\!\triangleright_i \Gamma' \triangleright W'$ follows. Condition (ii), (Time), is not possible because in the original sequence (A.1) above the testing component $\mathit{eureka}!\langle ok\rangle$ can not make a σ move, hence it cannot perform a timed reduction $\rightarrow\!\!\triangleright_\sigma$.

(2) Suppose now that $\Gamma_1 \triangleright W_1 \rightarrow\!\!\triangleright_\sigma \Gamma_1' \triangleright W_1'$. In this case we will use the testing context:

$$T = \sigma.(\tau.\mathit{eureka}!\langle ok\rangle + \mathit{fail}!\langle no\rangle)$$

where *eureka* and *fail* are fresh channels. Since $\Gamma_1 \triangleright W_1 \rightarrow\!\!\triangleright_\sigma \Gamma_1' \triangleright W_1'$ we also have $\Gamma_1 \triangleright W_1 \mid T \rightarrow\!\!\triangleright_\sigma \rightarrow\!\!\triangleright_i C_1$, where $C_1 \triangleright = \Gamma_1' \triangleright W' \mid \mathit{eureka}!\langle ok\rangle)$. Note that, since *fail* is a fresh channel, we have that $C_1 \Downarrow_{\mathit{eureka}}$ and $C_1 \not\Downarrow_{\mathit{fail}}$.

The contextuality of $\simeq$ gives that $\Gamma_1 \triangleright W_1 \mid T \simeq \Gamma_2 \triangleright W_2 \mid T$, so that we must have the series of reduction steps

$$\Gamma_2 \triangleright W_2 \mid T \rightarrow^* C_2 \tag{A.2}$$

where $C_1 \simeq C_2$. Because $C_1 \Downarrow_{eureka}$ and $C_1 \not\Downarrow_{fail}$, the same must be true of C_2. As $\Gamma'_1 \vdash eureka : \mathbf{idle}$, it follows that C_2 must take the form $\Gamma'_2 \triangleright W'_2 \mid eureka!\langle ok\rangle$. By Lemma 4.14 we have that $\Gamma'_1 \triangleright W'_1 \simeq \Gamma'_2 \triangleright W'_2$. It remains to establish that $\Gamma_2 \triangleright W_2 \rightarrow_i^* \rightarrow_\sigma \rightarrow_i^* \Gamma'_2 \triangleright W'_2$.

We proceed as in the previous proposition, by extracting out of (A.2) the contributions from $\Gamma_2 \triangleright W_2$; we know that because of the presence of the time delay in T, one time unit needs to pass before the broadcast along *eureka* is enabled in $\Gamma_2 \triangleright W_2 \mid T$; also, by maximal progress (Proposition 2.11), we know that such a broadcast must be fired before time passes. So (A.2) actually takes the form

$$\Gamma_2 \triangleright W_2 \mid T \rightarrow_i^* \Gamma' \triangleright W' \mid \ldots \rightarrow_\sigma \Gamma'' \triangleright W'' \mid \ldots \rightarrow_i^* \Gamma'_2 \triangleright W'_2 \mid eureka!\langle ok\rangle$$

Each individual reduction step can now be projected on to the first component, giving the required

$$\Gamma_2 \triangleright W_2 \rightarrow_i^* \Gamma \triangleright W \stackrel{\sigma}{\longmapsto} \Gamma' \triangleright W' \rightarrow_i^* \Gamma'_2 \triangleright W'_2$$

□

Proof of Proposition 4.18. The two implications are proved separately; first, let $\Gamma \triangleright W$ be a configuration such that $\Gamma \triangleright W \stackrel{c?v}{\Longmapsto} \Gamma' \triangleright W'$; that is, $\Gamma \triangleright W \Longmapsto \Gamma^{\text{pre}} \triangleright W^{pre} \stackrel{c?v}{\longmapsto} \Gamma^{\text{post}} \triangleright W^{\text{post}} \Longmapsto \Gamma' \triangleright W'$. Since $T_{c?v}$ does not contain any receiver, nor does $T^{\checkmark}_{c?v}$, we have the sequences of transitions $\Gamma \triangleright W \mid T_{c?v} \Longmapsto \Gamma^{\text{pre}} \triangleright W^{\text{pre}} \mid T_{c?v}$ and $\Gamma^{\text{post}} \triangleright W^{\text{post}} \mid T^{\checkmark}_{c?v} \Longmapsto \Gamma' \triangleright W' \mid T^{\checkmark}_{c?v}$.

Next we show that $\Gamma^{\text{pre}} \triangleright W^{\text{pre}} \mid T_{c?v} \stackrel{\tau}{\longmapsto} \Gamma^{\text{post}} \triangleright W^{\text{post}} \mid T^{\checkmark}_{c?v}$. Combined with the two (weak) transitions above, this gives the extensional transition $\Gamma \triangleright W \mid T_{c?v} \Longmapsto \Gamma \triangleright W' \mid T^{\checkmark}_{c?v}$, which can be rewritten as $\Gamma \triangleright W \mid T_{c?v} \rightarrow_i^* \Gamma \triangleright W' \mid T^{\checkmark}_{c?v}$.

Consider then the transition $\Gamma^{\text{pre}} \triangleright W^{pre} \stackrel{c?v}{\longmapsto} \Gamma^{\text{post}} \triangleright W^{\text{post}}$; this can only have been obtained by the intensional transition $\Gamma^{\text{pre}} \triangleright W^{\text{pre}} \xrightarrow{c?v} W'$, and the equality $\Gamma^{\text{post}} = \text{upd}_{c?v}(\Gamma^{\text{pre}})$. For the test $T_{c?v}$ we have the transition $\Gamma^{\text{pre}} \triangleright T_{c?v} \xrightarrow{c!v} T^{\checkmark}_{c?v}$; Now we can combine the two transitions together, using Rule (sync), and get $\Gamma^{\text{pre}} \triangleright W^{\text{pre}} \mid T_{c?v} \xrightarrow{c!v} W^{\text{post}} \mid T^{\checkmark}_{c?v}$; also, we know that $\Gamma^{\text{post}} = \text{upd}_{c?v}(\Gamma^{\text{pre}}) = \text{upd}_{c!v}(\Gamma^{\text{pre}})$, hence we can infer the required transition $\Gamma^{\text{pre}} \triangleright W^{\text{pre}} \mid T_{c?v} \stackrel{\tau}{\longmapsto} \Gamma^{\text{post}} \triangleright W^{\text{post}} \mid T^{\checkmark}_{c?v}$.

For the other implication, suppose that $\Gamma \triangleright W \mid T_{c?v} \rightarrow_i^* \Gamma' \triangleright W' \mid T^{\checkmark}_{c?v}$. This is possible only if, at some point in the sequence, the test component $T_{c?v}$ fired the broadcast along channel c; in fact, we have that the broadcast along channel *eureka* is guarded by a broadcast action in $T_{c?v}$, while it is guarded by a delay of δ_v instants of time in $T^{\checkmark}_{c?v}$. Also, by Maximal Progress (Proposition 2.11) the broadcast performed by $T_{c?v}$ must happen before time elapses; formally, we have the sequence of reductions

$$\Gamma \triangleright W \mid T_{c?v} \rightarrow_i^* \Gamma^{\text{pre}} \triangleright W^{\text{pre}} \mid T_{c?v} \rightarrow_i \Gamma^{\text{post}} \triangleright W^{\text{post}} \mid T^{\checkmark}_{c?v} \rightarrow_i^* \Gamma' \triangleright W' \mid T^{\checkmark}_{c?v}$$

Now note that the sequence of instantaneous reductions

$$\Gamma \triangleright W \mid T_{c?v} \rightarrow_i^* \Gamma^{\text{pre}} \triangleright W^{\text{pre}} \mid T_{c?v} \tag{A.3}$$

induces the extensional transition $\Gamma \triangleright W \Longmapsto \Gamma^{\text{pre}} \triangleright W^{\text{pre}}$. This can be proved using the facts that, for any channel environment Γ_x and channel d, whenever $\Gamma_x \triangleright T_{c?v} \overset{\tau}{\longmapsto} \Gamma'_x \triangleright T'$, then $T' = T_{c?v}$, and whenever $\Gamma_x \triangleright T_{c?v} \overset{\tau}{\Longmapsto} \Gamma'_x \triangleright T'$ then $T' \neq T_{c?v}$.

Similarly, we can prove that the weak reduction

$$\Gamma^{\text{post}} \triangleright W^{\text{post}} \mid T^{\checkmark}_{c?v} \rightarrow^*_i \Gamma' \triangleright W' \mid T^{\checkmark}_{c?v}$$

induces the extensional transition $\Gamma^{\text{post}} \triangleright W^{\text{post}} \Longmapsto \Gamma' \triangleright W'$.

It remains to show that we can infer the transition $\Gamma^{\text{pre}} \triangleright W^{\text{pre}} \overset{c?v}{\longmapsto} \Gamma^{\text{post}} \triangleright W^{\text{post}}$ from the reduction $\Gamma^{\text{pre}} \triangleright W^{\text{pre}} \mid T_{c?v} \rightarrow_i \Gamma^{\text{post}} \triangleright W^{\text{post}} \mid T^{\checkmark}_{c?v}$. Note that in $T_{c?v}$ we have a station which is ready to broadcast along channel c, while this is not true anymore in $T^{\checkmark}_{c?v}$. By performing a case analysis on the intensional transition which could have led to the reduction above, we find that the only possible case is that $\Gamma^{\text{pre}} \triangleright W^{\text{pre}} \mid T_{c?v} \xrightarrow{c!v} W^{\text{post}} \mid T^{\checkmark}_{c?v}$ and, more specifically, that $\Gamma^{\text{pre}} \triangleright W^{\text{pre}} \xrightarrow{c?v} W^{\text{post}}$ and $\Gamma^{\text{pre}} \triangleright T_{c?v} \xrightarrow{c!v} T^{\checkmark}_{c?v}$. Also, $\Gamma^{\text{post}} \triangleright = \text{upd}_{c!v}(\Gamma \triangleright^{\text{pre}})$. By an application of Rule (Input) in the extensional semantics, we get the required transition $\Gamma^{\text{pre}} \triangleright W^{\text{pre}} \overset{c?v}{\longmapsto} \Gamma^{\text{post}} \triangleright W^{\text{post}}$, which can be combined with the two weak transitions already derived, namely $\Gamma \triangleright W \Longmapsto \Gamma^{\text{pre}} \triangleright W^{\text{pre}}$ and $\Gamma^{\text{post}} \triangleright W^{\text{post}} \Longmapsto \Gamma' \triangleright W'$, to obtain $\Gamma \triangleright W \overset{c?v}{\Longmapsto} \Gamma' \triangleright W'$. □

Proof of Proposition 4.19. Suppose that $\Gamma \triangleright W \overset{\iota(c)}{\Longmapsto} \Gamma' \triangleright W'$. This can be rewritten as $\Gamma \triangleright W \Longmapsto \Gamma^{\text{pre}} \triangleright W^{\text{pre}} \overset{\iota(c)}{\longmapsto} \Gamma^{\text{post}} \triangleright W^{\text{post}} \Longmapsto \Gamma' \triangleright W'$. Since the only rule of the extensional semantics that could have been used to derive $\Gamma^{\text{pre}} \triangleright W^{\text{pre}} \overset{\iota(c)}{\longmapsto} \Gamma^{\text{post}} \triangleright W^{\text{post}}$ is (Idle), we obtain that $\Gamma^{\text{pre}} \triangleright W^{\text{pre}} = \Gamma^{\text{post}} \triangleright W^{\text{post}}$. Thus, we have $\Gamma \triangleright W \Longmapsto \Gamma^{\text{pre}} \triangleright W^{\text{pre}} = \Gamma^{\text{post}} \triangleright W^{\text{post}} \Longmapsto \Gamma' \triangleright W'$, or equivalently $\Gamma \triangleright W \overset{\Gamma' \triangleright W'}{\Longmapsto}$. In terms of the reduction semantics, this can be rewritten as $\Gamma \triangleright W \rightarrow^*_i \Gamma' \triangleright W'$.

By Corollary A.5 we know that $\Gamma \triangleright W \overset{\iota(c)}{\Longmapsto}$ implies $\Gamma \triangleright W \overset{\iota(c)}{\longmapsto} \Gamma \triangleright W$; therefore $\Gamma \vdash c : \textbf{idle}$. Now it is easy to see that we have the reduction $\Gamma \triangleright W \mid T_{\iota(c)} \rightarrow_i \Gamma \triangleright W \mid T^{\checkmark}_{\iota(c)} \rightarrow^*_i \Gamma' \triangleright W' \mid T_{\iota(c)}$, where the first reduction has been obtained by letting the predicate $\exp(c)$ be evaluated in $T_{\iota(c)}$, while the rest of the sequence can be derived using the facts that $\Gamma \triangleright W \rightarrow^*_i \Gamma' \triangleright W'$, and for any channel environment Γ_x we have that $\Gamma_x \triangleright T^{\checkmark}_{\iota(c)} \not\rightarrow \triangleright_i$, $\Gamma_x \triangleright T^{\checkmark}_{\iota(c)} \xrightarrow{c?v} T'$ implies $T' = T^{\checkmark}_{\iota(c)}$.

Conversely, suppose that $\Gamma \triangleright W \mid T_{\iota(c)} \rightarrow^i_* \Gamma' \triangleright W' \mid T^{\checkmark}_{\iota(c)}$. In this sequence of reductions, the evolution of the test component from $T_{\iota(c)}$ to $T^{\checkmark}_{\iota(c)}$ is possible only if eventually the exposure check on channel c is evaluated to true. That is, we have the sequence of reductions

$$\Gamma \triangleright W \mid T_{\iota(c)} \rightarrow^*_i \Gamma^{\text{pre}} \triangleright W^{\text{pre}} \mid T_{\iota(c)} \rightarrow_i \Gamma^{\text{post}} \triangleright W^{\text{post}} \mid T^{\checkmark}_{\iota(c)} \rightarrow^*_i \Gamma' \triangleright W' \mid T^{\checkmark}_{\iota(c)}$$

where $\Gamma^{\text{pre}} \vdash c : \textbf{idle}$.

Since the evaluation of the exposure check in the reduction $\Gamma^{\text{pre}} \triangleright W^{\text{pre}} \mid T_{\iota(c)} \rightarrow_i \Gamma^{\text{post}} \triangleright W^{\text{post}} \mid T^{\checkmark}_{\iota(c)}$ corresponds to a τ-intensional transition which affects only the system term $T_{\iota(c)}$, that is $\Gamma^{\text{pre}} \triangleright T_{\iota(c)} \xrightarrow{\tau} T^{\checkmark}_{\iota(c)}$, Proposition 2.13(1) ensures that $W^{\text{post}} = W^{\text{pre}}$, and $\Gamma^{\text{post}} = \text{upd}_{\tau}(\Gamma^{\text{pre}}) = \Gamma^{\text{pre}}$. Using the facts that $\Gamma^{\text{pre}} \triangleright W^{\text{pre}} = \Gamma^{\text{post}} \triangleright W^{\text{post}}$ and $\Gamma^{\text{pre}} \vdash c : \textbf{idle}$, we can apply Rule (Idle) of the extensional semantics and infer the transition $\Gamma^{\text{pre}} \triangleright W^{\text{pre}} \overset{\iota(c)}{\longmapsto} \Gamma^{\text{post}} \triangleright W^{\text{post}}$.

Next, note that for any configuration Γ_x, we have that $\Gamma_x \triangleright T_{\iota(c)} \xrightarrow{d?v} T'$ implies $T' = T_{\iota(c)}$, and $\Gamma_x \triangleright T_{\iota}(c) \rightarrow_i \Gamma'_x \triangleright T'$ implies $T' \neq T_{\iota(c)}$. Similar results hold for the system term $T^{\checkmark}_{\iota(c)}$. Using these facts, it is not difficult can derive the extensional transition $\Gamma \triangleright W \Longmapsto \Gamma^{\text{pre}} \triangleright W^{\text{pre}}$ from the sequence

of reductions $\Gamma \triangleright W \mid T_{\iota(c)} \rightarrow_i^* \Gamma^{\text{pre}} \triangleright W^{\text{pre}}$, and the transition $\Gamma^{\text{post}} \triangleright W^{\text{post}} \Longmapsto \Gamma' \triangleright W'$ from the sequence of reductions $\Gamma^{\text{post}} \triangleright W^{\text{post}} \mid T^{\surd}_{\iota(c)} \rightarrow_i^* \Gamma' \triangleright W' \mid T^{\surd}_{\iota(c)}$.

Thus we have proved that $\Gamma \triangleright W \Longmapsto \Gamma^{\text{pre}} \triangleright W^{\text{pre}} \overset{\iota(c)}{\longmapsto} \Gamma^{\text{post}} \triangleright W^{\text{post}} \Longmapsto \Gamma' \triangleright W'$, or equivalently $\Gamma \triangleright W \overset{\iota(c)}{\Longmapsto} \Gamma' \triangleright W'$. □

Proof of Proposition 4.20. For any value w, let T_w be the system term

$$T_w = \nu d : (0, \cdot).(([w = v]d!\langle ok\rangle, \mathsf{nil}) + \mathit{fail}!\langle no\rangle \mid \sigma.[\exp(d)]\mathit{eureka}!\langle ok\rangle, \mathsf{nil})$$

Suppose that $\Gamma \triangleright W \overset{\gamma(c,v)}{\Longmapsto} \Gamma' \triangleright W'$. In particular, we have that $\Gamma \triangleright W \Longmapsto \Gamma^{\text{pre}} \triangleright W^{\text{pre}} \overset{\gamma(c,v)}{\longmapsto} \Gamma^{\text{post}} \triangleright W^{\text{post}} \Longmapsto \Gamma' \triangleright W'$. From the transition $\Gamma^{\text{pre}} \triangleright W^{\text{pre}} \overset{\gamma(c,v)}{\longmapsto} \Gamma^{\text{post}} \triangleright W^{\text{post}}$ we get that $\Gamma^{\text{pre}} = (1, v)$, and $\Gamma^{\text{pre}} \triangleright W^{\text{pre}} \xrightarrow{\sigma} W^{\text{post}}$. In particular, note that $\Gamma^{\text{pre}} \vdash c : \mathbf{exp}$, hence $\Gamma \triangleright^{\text{pre}} W^{\text{pre}} \mid T_{\iota(c,v)}$ is well formed. Note also that $\Gamma_x \triangleright T_{\gamma(c,v)} \not\rightarrow_i$ for any environment Γ_x with $\Gamma_x \vdash c : \mathbf{exp}$, and that $\Gamma_x \triangleright T_{\gamma(c,v)} \xrightarrow{c?v} T'$ implies that $T' = T_{\gamma(c,v)}$. Also, since $\Gamma^{\text{pre}}(c) = (1, v)$, we obtain the transition $\Gamma^{\text{pre}} \triangleright T_{\iota(c)} \xrightarrow{\sigma} T_v$. Finally, note that, for any channel environment Γ_x we also have the transition $\Gamma_x \triangleright T_v \xrightarrow{\tau} T^{\surd}_{\gamma(c,v)}$. Using these facts, we can build the sequence of transitions

$$\Gamma \triangleright W \mid T_{\gamma(c,v)} \rightarrow_i^* \Gamma^{\text{pre}} \triangleright W^{\text{pre}} \mid T_{\gamma(c,v)} \rightarrow_\sigma \Gamma^{\text{post}} \triangleright W^{\text{post}} \mid T_v \rightarrow_i^* \Gamma' \triangleright W' \mid T_v \rightarrow_i \Gamma' \triangleright W' \mid T^{\surd}_{\gamma(c,v)}$$

Now suppose that $\Gamma \triangleright W T_{\gamma(c,v)} \rightarrow_i^* \rightarrow_\sigma \rightarrow_i^* \Gamma \triangleright W' \mid T_{\gamma(c,v)}$; we need to show that $\Gamma \triangleright W \overset{\gamma(c,v)}{\Longmapsto} \Gamma' \triangleright W'$. Note that, in order for the testing component $T_{\gamma(c,v)}$ to evolve into $T^{\surd}_{\gamma(c,v)}$, then

(1) when the first time instant passes, the test evolves into T_w for some value w; this is because in $T^{\surd}_{\gamma(c,v)}$ the active receiver along channel c has vanished, and in CCCP active receivers along a channel c can only disappear after a timed reduction has been performed, and only if the state of channel c changes from exposed to idle,
(2) at some point, in the remaining of the computation, the matching construct $[w = v]$ is evaluated in T_w, leading to the test component to evolve in $T^{\surd}_{\gamma(c,v)}$. Note that the matching construct $[w = v]$ cannot be evaluated to false, as this would cause the test component to evolve to a system term different from $T^{\surd}_{\gamma(c,v)}$. Therefore, $w = v$, and more specifically $T_w = T_v$.
(3) The evaluation of the matching construct $[v = v]$ to true is modelled as an τ-intensional action, hence it does not affect the tested component W.

Formally, we have a sequence of reductions

$$\begin{aligned} & \Gamma \triangleright W \mid T_{\gamma(c,v)} \rightarrow_i^* \Gamma^{\text{pre}} \triangleright W^{\text{pre}} \mid T_{\gamma(c,v)} && \rightarrow_\sigma \\ \rightarrow_\sigma \quad & \Gamma^{\text{post}} \triangleright W^{\text{post}} \mid T_v \rightarrow_i^* \Gamma'' \triangleright W'' \mid T_v && \rightarrow_i \\ \rightarrow_i \quad & \Gamma'' \triangleright W'' \mid T^{\surd}_{\gamma(c,v)} \rightarrow_i^* \Gamma' \triangleright W' \mid T^{\surd}_{\gamma(c,v)} \end{aligned}$$

where $\Gamma^{\text{pre}}(c) = (1, v)$.

Let T be either $T_{\gamma c}, T_v$ or $T^{\surd}_{\gamma c,v}$, and let Γ_x be an arbitrary channel environment; note that we have that $\Gamma_x \triangleright T \xrightarrow{d?v} T'$ implies $T' = T$, and $\Gamma_x \triangleright T \rightarrow_i \Gamma'_x \triangleright T'$ implies that $T' \neq T$. Using these facts, it is not difficult to derive the transitions

(a) $\Gamma \triangleright W \Longmapsto \Gamma^{\text{pre}} \triangleright W^{\text{pre}}$,
(b) $\Gamma^{\text{post}} \triangleright W^{\text{post}} \Longmapsto \Gamma'' \triangleright W''$,
(c) $\Gamma'' \triangleright W'' \Longmapsto \Gamma' \triangleright W'$

Thus, we only need to show that $\Gamma^{\text{pre}} \triangleright W^{\text{pre}} \overset{\gamma(c,v)}{\longmapsto} \Gamma^{\text{post}} \triangleright W^{\text{post}}$. The timed reduction $\Gamma^{\text{pre}} \triangleright W^{\text{pre}} \mid T_{\gamma(c,v)} \twoheadrightarrow_{\sigma} \Gamma^{\text{post}} \triangleright W^{\text{post}} \mid T_v$ can only be inferred if $\Gamma^{\text{pre}} \triangleright W^{\text{pre}} \xrightarrow{\sigma} W^{\text{post}}$, $\Gamma^{\text{pre}} \triangleright T_{\gamma(c,v)} \xrightarrow{\sigma} T_v$ and $\Gamma^{\text{post}} = \text{upd}_{\sigma}(\Gamma^{\text{pre}})$. Also, note that the only possibility for inferring the transition $\Gamma^{\text{pre}} \triangleright T_{\gamma(c,v)} \xrightarrow{\sigma} T_v$ is by using an instance of Rule (EndRcv) (where the channel environment contains value v at channel c); therefore, we obtain that $\Gamma^{\text{pre}}(c) = (1, v)$.

We have proved that $\Gamma^{\text{pre}}(c) = (1, v)$, $\Gamma^{\text{pre}} \triangleright W^{\text{pre}} \xrightarrow{\sigma} W^{\text{post}}$ and $\Gamma^{\text{post}} = \text{upd}_{\sigma}(\Gamma^{\text{pre}})$; therefore, we can apply Rule (Deliver) to infer that $\Gamma^{\text{pre}} \triangleright W^{\text{pre}} \overset{\gamma(c,v)}{\longmapsto} \Gamma^{\text{post}} \triangleright W^{\text{post}}$, as we wanted to show. By combining this transition with the weak transitions listed in (a), (b), (c), above, we obtain the required $\Gamma \triangleright W \overset{\gamma(c,v)}{\Longmapsto} \Gamma' \triangleright W'$. □