Journal of Computer Security 28 (2020) 71–127
DOI 10.3233/JCS-191366
IOS Press

On the semantics of communications when verifying equivalence properties

Kushal Babel [a], Vincent Cheval [b,*] and Steve Kremer [b]

[a] *Cornell University, US*
E-mail: babel@cs.cornell.edu
[b] *Inria Nancy Grand-Est, Loria, France*
E-mails: vincent.cheval@inria.fr, steve.kremer@inria.fr

Abstract. Symbolic models for security protocol verification were pioneered by Dolev and Yao in their seminal work. Since then, although inspired by the same ideas, many variants of the original model were developed. In particular, a common assumption is that the attacker has complete control over the network and can therefore intercept any message. This assumption has been interpreted in slightly different ways depending on the particular models: either any protocol output is directly routed to the adversary, or communications may be among any two participants, including the attacker – the scheduling between which exact parties the communication happens is left to the attacker. This difference may seem unimportant at first glance and, depending on the verification tools, either one or the other semantics is implemented. We show that, unsurprisingly, they indeed coincide for reachability properties. However, for indistinguishability properties, we prove that these two interpretations lead to incomparable semantics. We also introduce and study a new semantics, where internal communications are allowed but messages are always eavesdropped by the attacker. This new semantics yields strictly stronger equivalence relations. Moreover, we identify two subclasses of protocols for which the three semantics coincide. Finally, we implemented verification of trace equivalence for each of the three semantics in the DeepSec tool and compare their performances on several classical examples.

Keywords: Cryptographic protocols, symbolic models, verification, semantics, equivalence properties

1. Introduction

Automated, symbolic analysis of security protocols, based on the seminal ideas of Dolev and Yao, comes in many variants. All of these models however share a few fundamental ideas:

- messages are represented as abstract terms,
- adversaries are computationally unbounded, but may manipulate messages only according to predefined rules (this is sometimes referred to as the perfect cryptography assumption), and
- the adversary completely controls the network.

In this paper we will revisit this last assumption. Looking more precisely at different models we observe that this assumption may actually slightly differ among the models. The fact that the adversary controls the network is supposed to represent a *worst case* assumption.

In some models this assumption translates to the fact that every protocol output is sent to the adversary, and every protocol input is provided by the adversary. This is the case in the original Dolev Yao model and also in the models underlying several tools, such as AVISPA [8], Scyther [20], Tamarin [27], Millen

[*] Corresponding author. E-mail: vincent.cheval@inria.fr.

and Shmatikov's constraint solver [24], and the model used in Paulson's inductive approach [25]. We will refer to this choice of semantics as the *private* semantics, as internal communications are only allowed on private channels.

Some other models, such as those based on process algebras, e.g. work based on CSP [26], the Spi [3] and applied pi calculus [1], but also the strand space model [28], consider a slightly different communication model: any two agents may communicate. Scheduling whether communication happens among two honest participants, or a honest participant and the attacker is under the attacker's control. We will refer to this choice of semantics as the *classical* semantics, as it corresponds to what is generally used in process calculi.

When considering *reachability properties*, these two communication models indeed coincide: intuitively, any internal communication could go through the adversary who acts as a relay and increases his knowledge by the transmitted message. However, when considering *indistinguishability properties*, typically modelled as process equivalences, these communication models diverge. Interestingly, when forbidding internal communication, i.e., forcing all communication to be relayed by the attacker, we may weaken the attacker's distinguishing power. This observation may seem counter-intuitive at first. However, executing a (non-observable) internal communication may enable actions that are otherwise only available after an observable input. These actions may then provide additional capabilities for simulating the other process.

In many recent work privacy properties have been modelled using process equivalences, see for instance [6,21,22]. The number of tools able to verify such properties is also increasing [12–14,16,18,29]. For instance, the AKISS [13] and SAT-EQUIV [18] tools do not allow any direct communication on public channels, while the APTE [14] and DeepSec [16] tools allow for internal communications. One motivation for disallowing direct communication is that it allows for more efficient verification (as less actions need to be considered and the number of interleavings to be considered is smaller).

Our contributions. We have formalised three semantics in the applied pi calculus which differ by the way communication is handled:

- the *classical* semantics (as in the original applied pi calculus) allows both internal communication among honest participants and communication with the adversary;
- the *private* semantics allows internal communication only on private channels while all communication on public channels is routed through the adversary;
- the *eavesdropping* semantics which allows internal communication, but as a side-effect adds the transmitted message to the adversary's knowledge.

For each of the new semantics we define may-testing and observational equivalences. We also define corresponding labelled semantics and trace equivalence and bisimulation relations (which may serve as proof techniques).

We show that, as expected, the three semantics coincide for reachability properties. For equivalence properties we show that the classical and private semantics yield incomparable equivalences, while the eavesdropping semantics yields strictly stronger equivalence relations than both other semantics. The results are summarized in Fig. 4.

An interesting question is whether these semantics coincide for specific subclasses of processes. We note that the processes that witness the differences in the semantics do not use replication, private channels, nor terms other than names, and no equational theory. Moreover, all except one of these examples only use trivial *else* branches (of the form else 0); the use of a non-trivial else branch can even be avoided by allowing a single free symbol.

We first study different notions of determinate processes: in the context of the applied pi calculus, Cheval et al. [15] have for instance shown that observational, testing, trace equivalence and labelled bisimulation coincide for this class of processes (for the classic semantics). We will show that this is actually the case for all semantics and show, among others that the private and eavesdropping semantics do coincide on these equivalences, and imply them for the classic semantics. We consider several specific subclasses of determinate processes when we bound the number of sessions. In particular, we show that all equivalences and semantics coincide for the class of *strong action determinate* processes. This class is of practical importance as this condition is checked in the AKISS and DeepSec tools to enable partial order reduction optimizations [10]. These optimizations provide spectacular speed-ups, but they were designed and shown correct only in the private semantics. Showing that all three semantics coincide for strong action determinate processes lifts the benefit of these optimizations to the other semantics. The results on subclasses of determinate processes are summarized in Fig. 8.

We also identify a syntactic class of processes, that we call *I/O-unambiguous*. For this class we forbid communication on private channels, communication of channel names and an output may not be sequentially followed by an input on the same channel directly, or with only conditionals in between. Note however that I/O-unambiguous processes, unlike most determinate processes, do allow outputs and inputs on the same channel in parallel. We show that for this class the eavesdropping semantics (which is the most strict relation) coincides with the private one (which is the most efficient for verification).

Finally, we extended the DeepSec tool to support verification of trace equivalence for the three semantics. Verifying existing protocols in the DeepSec example repository we verified that the results, fortunately, coincided for each of the semantics. We also made slight changes to the encodings, renaming some channels, to make them I/O-unambiguous. Interestingly, using different channels, significantly increased the performance of the tool. Finally, we also observed that, as expected, the private semantics yields more efficient verification. The results of our experiments are summarized in Section 5.

A preliminary version of this work appeared in [9]. In contrast to [9], this work contains full proofs of all results, new results for several subclasses of processes, giving a detailed comparison of the different semantics and equivalences, as well as an implementation of all three semantics in the DeepSec tool, together with an experimental evaluation.

Outline. In Section 2 we define the three semantics we consider. In Section 3 we present our main results on comparing these semantics. We present subclasses for which (some) semantics coincide in Section 4 and compare the performances when verifying protocols for different semantics using DeepSec in Section 5, before concluding in Section 6.

2. Model

The *applied pi calculus* [1] is a variant of the pi calculus that is specialised for modelling cryptographic protocols. Participants in a protocol are modelled as processes and the communication between them is modelled by message passing on channels. In this section, we describe the syntax and semantics of the applied pi calculus as well as the two new variants that we study in this paper.

2.1. Syntax

We consider an infinite set $\mathcal{N}$ of names of *base type* and an infinite set $\mathcal{C}h$ of names of *channel type*. We also consider an infinite set of variables $\mathcal{X}$ of base type and channel type and a signature $\mathcal{F}$ consisting

$$\begin{array}{rll@{\qquad}rll}
P, Q := & 0 & \text{plain processes} & A, B := & P & \text{extended processes}\\
& P \mid Q & & & A \mid B & \\
& !P & & & \nu n.A & \\
& \nu n.P & & & \nu x.A & \\
& \mathsf{if}\ u = v\ \mathsf{then}\ P\ \mathsf{else}\ Q & & & \{^{u}/_{x}\} & \\
& \mathsf{in}^{\theta}(c, x).P & & & \omega c & \\
& \mathsf{out}^{\theta}(c, u).P & & & & \\
& \mathsf{eav}(c, x).P & & & &
\end{array}$$

where u and v are base type terms, n is a name, x is a variable and c is a name or variable of channel type, θ is a tag, *i.e.* $\theta \in \{\mathsf{ho}, \mathsf{at}\}$.

Fig. 1. Syntax of processes.

of a finite set of *function symbols*. We rely on a sort system for terms. In particular, the sort base type differs from the sort channel type. Moreover, any function symbol can only be applied to and returns base type terms. We define *terms* as names, variables and function symbols applied to other terms. Given $N \subseteq \mathcal{N}$, $X \subseteq \mathcal{X}$ and $F \subseteq \mathcal{F}$, we denote by $\mathcal{T}(F, X, N)$ the sets of terms built from X and N by applying function symbols from F. We denote $v(t)$ the sets of variables occurring in t. We say that t is *ground* if $v(t) = \emptyset$. We describe the behaviour of cryptographic primitives by the means of an *equational theory* E that is a relation on terms closed under substitutions of terms for variables and closed under one-to-one renaming. Given two terms u and v, we write $u =_{\mathsf{E}} v$ when u and v are equal modulo the equational theory.

In the original syntax of the applied pi calculus, there is no distinction between an output (resp. input) from a protocol participant and from the environment, also called the attacker. In this paper however, we will make this distinction in order to concisely present our new variants of the semantics. Therefore, we consider two *process tags* ho and at that respectively represent honest and attacker actions. The syntax of *plain processes* and *extended processes* is given in Fig. 1.

The process $\mathsf{out}^{\theta}(c, u)$ represents the output by θ of the message u on the channel c. The process $\mathsf{in}^{\theta}(c, x)$ represents an input by θ on the channel c. The input message will instantiate the variable x. The process $\mathsf{eav}(c, x)$ models the capability of the attacker to eavesdrop a communication on channel c. The process $!P$ represents the replication of the process P, *i.e.* unbounded number of copies of P. The process $P \mid Q$ represents the parallel composition of P and Q. The process $\nu n.P$ (resp. $\nu x.A$) is the restriction of the name n in P (resp. variable x in A). The process $\mathsf{if}\ u = v\ \mathsf{then}\ P\ \mathsf{else}\ Q$ is the conditional branching under the equality test $u = v$. The process ωc records that a private channel c has been opened, i.e., it has been sent on a public or previously opened channel. Finally, the substitution $\{^{u}/_{x}\}$ is an active substitution that replaces the variable x with the term u of base type.

We say that a process P (resp. extended process A) is an *honest process* (resp. *honest extended process*) when all inputs and outputs in P (resp. A) are tagged with ho and when P (resp. A) does not contain eavesdropping processes and ωc. We say that a process P (resp. extended process A) is an *attacker process* (resp. *attacker extended process*) when all inputs and outputs in P (resp. A) are tagged with at.

As usual, names and variables have scopes which are delimited by restrictions, inputs and eavesdrops. We denote $\mathit{fv}(A)$, $\mathit{bv}(A)$, $\mathit{fn}(A)$, $\mathit{bn}(A)$ the sets of free variables, bound variables, free names and bound names respectively in A. Moreover, we denote by $\mathit{oc}(A)$ the sets of terms c of channel type opened in A, *i.e.* that occurs in a process ωc. We say that an extended process A is closed when all variables in

A are either bound or defined by an active substitution in A. We define an *evaluation context* $C[\_]$ as an extended process with a hole instead of an extended process. As for processes, we define an *attacker evaluation context* as an evaluation context where all outputs and inputs in the context are tagged with at.

Note that our syntax without the eavesdropping process, opened channels and tags correspond exactly to the syntax of the original applied pi calculus.

Lastly, we consider the notion of *frame* that are extended processes built from 0, parallel composition, name and variable restrictions and active substitution. Given a frame φ, we consider the domain of φ, denoted $\mathrm{dom}(\varphi)$, as the set of free variables in φ that are defined by an active substitution in φ. Given an extended process A, we define the frame of A, denoted $\phi(A)$, as the process A where we replace all plain processes by 0. Finally, we write $\mathrm{dom}(A)$ as syntactic sugar for $\mathrm{dom}(\phi(A))$.

2.2. Operational semantics

In this section, we define the three semantics that we study in this paper, namely:

- the *classical semantics* from the applied pi calculus, where internal communication can occur on both public and private channels;
- the *private semantics* where internal communication can only occur on private channels; and
- the *eavesdropping semantics* where the attacker is able to eavesdrop on a public channel.

We first define the *structural equivalence* between extended processes, denoted $\equiv$, as the smallest equivalence relation on extended processes that is closed under renaming of names and variables, closed by application of evaluation contexts, that is associative and commutative w.r.t. $|$, and such that:

$$\begin{array}{ccc}
A \equiv A \mid 0 & !P \equiv !P \mid P & \nu n.0 \equiv 0 \\
\nu i.\nu j.A \equiv \nu j.\nu i.A & \nu x.\{^u/_x\} \equiv 0 & \{^u/_x\} \mid A \equiv \{^u/_x\} \mid A\{^u/_x\}
\end{array}$$

$$\begin{array}{lll}
A \mid \nu i.B \equiv \nu i.(A \mid B) & \text{when } i \notin \mathit{fv}(A) \cup \mathit{fn}(A) & \omega c \equiv \omega c \mid \omega c \\
\{^u/_x\} \equiv \{^v/_x\} & \text{when } u =_{\mathsf{E}} v &
\end{array}$$

The three operational semantics of extended processes are defined by the structural equivalence and by three respective *internal reductions*, denoted $\rightarrow_{\mathsf{c}}$, $\rightarrow_{\mathsf{p}}$ and $\rightarrow_{\mathsf{e}}$. These three reductions are the smallest relations on extended processes that are closed under application of evaluation context, structural equivalence and such that:

$$\begin{array}{ll}
\text{if } u = v \text{ then } P \text{ else } Q \xrightarrow{\tau}_s P \text{ where } u =_{\mathsf{E}} v \text{ and } s \in \{\mathsf{c}, \mathsf{p}, \mathsf{e}\} & \text{THEN} \\
\text{if } u = v \text{ then } P \text{ else } Q \xrightarrow{\tau}_s Q & \text{ELSE} \\
\qquad\qquad \text{where } u, v \text{ ground}, u \neq_{\mathsf{E}} v \text{ and } s \in \{\mathsf{c}, \mathsf{p}, \mathsf{e}\} & \\
& \\
\mathsf{out}^{\theta}(c, u).P \mid \mathsf{in}^{\theta'}(c, x).Q \xrightarrow{\tau}_{\mathsf{c}} P \mid Q\{^u/_x\} & \text{COMM}
\end{array}$$

$$\begin{array}{lr}
\nu c.(\mathsf{out}^{\theta}(c,u).P \mid \mathsf{in}^{\theta'}(c,x).Q \mid R) \xrightarrow{\tau}_{s} \nu c.(P \mid Q\{^{u}/_{x}\} \mid R) & \textsc{C-Priv} \\
\qquad\qquad\qquad\qquad \text{where } c \notin oc(R) \text{ and } s \in \{\mathsf{p},\mathsf{e}\} & \\
\mathsf{out}^{\theta}(c,u).P \mid \mathsf{in}^{\theta'}(c,x).Q \xrightarrow{\tau}_{s} P \mid Q\{^{u}/_{x}\} & \textsc{C-Env} \\
\qquad\qquad \mathsf{at} \in \{\theta,\theta'\}, u \text{ is of base type and } s \in \{\mathsf{p},\mathsf{e}\} & \\
\mathsf{out}^{\theta}(c,d).P \mid \mathsf{in}^{\theta'}(c,x).Q \xrightarrow{\tau}_{s} P \mid Q\{^{d}/_{x}\} \mid \omega d & \textsc{C-Open} \\
\qquad\qquad \mathsf{at} \in \{\theta,\theta'\}, d \text{ is of channel type and } s \in \{\mathsf{p},\mathsf{e}\} & \\
\mathsf{out}^{\mathsf{ho}}(c,u).P \mid \mathsf{in}^{\mathsf{ho}}(c,x).Q \mid \mathsf{eav}(c,y).R \xrightarrow{\tau}_{\mathsf{e}} P \mid Q\{^{u}/_{x}\} \mid R\{^{u}/_{y}\} & \textsc{C-Eav} \\
\qquad\qquad\qquad\qquad \text{where } u \text{ is of base type} & \\
\mathsf{out}^{\mathsf{ho}}(c,d).P \mid \mathsf{in}^{\mathsf{ho}}(c,x).Q \mid \mathsf{eav}(c,y).R \xrightarrow{\tau}_{\mathsf{e}} P \mid Q\{^{d}/_{x}\} \mid R\{^{d}/_{y}\} \mid \omega d & \textsc{C-OEav} \\
\qquad\qquad\qquad\qquad \text{where } d \text{ is of channel type} &
\end{array}$$

We emphasise that the application of the rule is closed under application of arbitrary evaluation contexts. In particular the context may restrict channels, *e.g.* the rule C-OPEN may be used under the context $\nu c.\_$ resulting in a private channel c, but with the attacker input/output being in the scope of this restriction. It follows from the definition of evaluation contexts that the resulting processes are always well defined. We denote by $\Rightarrow_s$ the reflexive, transitive closure of $\xrightarrow{\tau}_{s}$ for $s \in \{\mathsf{c},\mathsf{p},\mathsf{e}\}$. We note that the classical semantics $\xrightarrow{\tau}_{\mathsf{c}}$ is independent of the tags θ, θ', the eavesdrop actions and the ωc processes.

Example 1. Consider the process

$$A = (\nu d.\mathsf{out}^{\theta}(c,d).\mathsf{in}^{\theta}(d,x).P) \mid (\mathsf{in}^{\theta'}(c,y).\mathsf{out}^{\theta'}(y,t).Q)$$

where d is a channel name and t a term of base type. Suppose $\theta = \theta' = \mathsf{ho}$ then we have that communication is only possible in the classical semantics (using twice the COMM rule):

$$\begin{aligned}
A &\xrightarrow{\tau}_{\mathsf{c}} \nu d.(\mathsf{in}^{\theta}(d,x).P \mid \mathsf{out}^{\theta'}(d,t).Q\{^{d}/_{y}\}) \\
&\xrightarrow{\tau}_{\mathsf{c}} \nu d.(P\{^{t}/_{x}\} \mid Q\{^{d}/_{y}\})
\end{aligned}$$

while no transitions are available in the two other semantics. To enable communication in the eavesdropping semantics we need to explicitly add eavesdrop actions. Applying the rules C-OEAV and C-EAV we have that

$$\begin{aligned}
A \mid \mathsf{eav}(c,z_1).\mathsf{eav}(z_1,z_2).R &\xrightarrow{\tau}_{\mathsf{e}} \nu d.(\mathsf{in}^{\theta}(d,x).P \mid \mathsf{out}^{\theta'}(d,t).Q\{^{d}/_{y}\} \\
&\qquad \mid \mathsf{eav}(d,z_2).R\{^{d}/_{z_1}\} \mid \omega d) \\
&\xrightarrow{\tau}_{\mathsf{e}} \nu d.(P\{^{t}/_{x}\} \mid Q\{^{d}/_{y}\} \mid R\{^{d}/_{z_1}\}\{^{t}/_{z_2}\} \mid \omega d)
\end{aligned}$$

We note that the first transition adds the information ωd to indicate that d is now available to the environment.

Finally, if we consider that $\mathsf{at} \in \{\theta, \theta'\}$ then internal communication on a public channel is possible and, using rules C-OPEN and C-ENV we obtain for $s \in \{\mathsf{p},\mathsf{e}\}$ that

$$\begin{aligned}
A &\xrightarrow{\tau}_{s} \nu d.(\mathsf{in}^{\theta}(d,x).P \mid \mathsf{out}^{\theta'}(d,t).Q\{^{d}/_{y}\} \mid \omega d) \\
&\xrightarrow{\tau}_{s} \nu d.(P\{^{t}/_{x}\} \mid Q\{^{d}/_{y}\} \mid \omega d)
\end{aligned}$$

2.3. Reachability and behavioural equivalences

We are going to compare the relation between the three semantics for the two general kind of security properties, namely *reachability properties* encoding security properties such as secrecy, authentication, and *equivalence properties* encoding properties such as anonymity, unlinkability, strong secrecy, and receipt freeness. Intuitively, reachability properties encode that a process cannot reach some bad state. Equivalences define the fact that no attacker can distinguish two processes. This was originally defined by the *(may)-testing equivalence* [3] in the spi-calculus. An alternate equivalence, which was considered in the applied pi calculus [1], is observational equivalence.

Reachability properties can simply be encoded by verifying the capability of a process to perform an output on a given channel. We define $A \Downarrow_c^{s,\theta}$ to hold when $A \Rightarrow_s C[\mathsf{out}^\theta(c,t).P]$ for some evaluation context C that does not bind c, some term t and some plain process P, and $A \Downarrow_c^s$ to hold when $A \Downarrow_c^{s,\theta}$ for some $\theta \in \{\mathsf{at}, \mathsf{ho}\}$. For example the secrecy of s in the process $\nu s.A$ can be encoded by checking whether for all attacker plain process I, we have that

$$I \mid \nu s.(A \mid \mathsf{in}^{\mathsf{ho}}(c,x).\mathsf{if}\ x = s\ \mathsf{then}\ \mathsf{out}^{\mathsf{ho}}(\mathsf{bad}, s)) \not\Downarrow_{\mathsf{bad}}^{s,\mathsf{ho}}$$

where $\mathsf{bad} \notin \mathit{fn}(A)$.

Authentication properties are generally expressed as correspondence properties between events annotating processes, see e.g. [11]. A correspondence property between two events begin and end, denoted begin $\Leftarrow$ end, requires that the event end is preceded by the event begin on every trace. A possible encoding of this correspondence property consists in first replacing all instances of the events in A by outputs $\mathsf{out}^{\mathsf{ho}}(ev, \mathsf{begin})$ and $\mathsf{out}^{\mathsf{ho}}(ev, \mathsf{end})$ where $ev \notin \mathit{fn}(A) \cup \mathit{bn}(A)$. This new process A' can then be put in parallel with a cell *Cell* that reads on the channel ev and stores any new value unless the value is end and the current stored value in the cell is not begin. In such a case, the cell will output on the channel bad. The correspondance property can therefore be encoded by checking whether for all attacker plain process I, we have that $I \mid \nu ev.(A' \mid \mathit{Cell}) \not\Downarrow_{\mathsf{bad}}^{s,\mathsf{ho}}$.

We say that an attacker evaluation context $C[\_]$ is c-closing for an extended process A if $\mathit{fv}(C[A]) = \emptyset$. For $s \in \{\mathsf{p}, \mathsf{e}\}$, we say that $C[\_]$ is s-closing for A if it is c-closing for A, variables and names are bound only once in $C[\_]$ and for all channels $c \in \mathit{bn}(C[\_]) \cap \mathit{fn}(A)$, if the scope of c includes _ then the scope of c also includes ωc.

We next introduce the two main notions of behavioural equivalences: may testing and observational equivalence.

Definition 1 ((May-)Testing equivalences $\approx_m^{\mathsf{c}}, \approx_m^{\mathsf{p}}, \approx_m^{\mathsf{e}}$)**.** Let $s \in \{\mathsf{c}, \mathsf{p}, \mathsf{e}\}$. Let A and B two closed honest extended processes such that $\mathrm{dom}(A) = \mathrm{dom}(B)$. We say that $A \approx_m^s B$ if for all attacker evaluation contexts $C[\_]$ s-closing for A and B, for all channels c, we have that $C[A] \Downarrow_c^s$ if and only if $C[B] \Downarrow_c^s$.

Definition 2 (Observational equivalences $\approx_o^{\mathsf{c}}, \approx_o^{\mathsf{p}}, \approx_o^{\mathsf{e}}$)**.** Let $s \in \{\mathsf{c}, \mathsf{p}, \mathsf{e}\}$. Let A and B two closed extended processes such that $\mathrm{dom}(A) = \mathrm{dom}(B)$. We say that $A \approx_o^s B$ if $\approx_o^s$ is the largest equivalence relation such that:

- $A \Downarrow_c^s$ implies $B \Downarrow_c^s$;
- $A \xrightarrow{\tau}_s A'$ implies $B \overset{\epsilon}{\Rightarrow}_s B'$ and $A' \approx_o^s B'$ for some B';
- $C[A] \approx_o^s C[B]$ for all attacker evaluation contexts $C[\_]$ s-closing for A and B.

$$A \mathrel{\hat{=}} \nu d.\mathsf{out}^{\mathsf{ho}}(d,a) \mid !\mathsf{in}^{\mathsf{ho}}(d,x).\mathsf{out}^{\mathsf{ho}}(d,h(x)) \mid \mathsf{in}^{\mathsf{ho}}(d,y).\mathsf{out}^{\mathsf{ho}}(c,y)$$
$$B \mathrel{\hat{=}} \nu e.\mathsf{out}^{\mathsf{ho}}(e,a) \mid \mathsf{in}^{\mathsf{ho}}(e,z).A \mid \mathsf{in}^{\mathsf{ho}}(e,z).\nu s.\mathsf{out}^{\mathsf{ho}}(c,s)$$

Fig. 2. Processes A and B such that $A \approx_m^s B$, but $A \not\approx_o^s B$ and $A \not\approx_t^s B$ for $s \in \{\mathsf{c}, \mathsf{e}, \mathsf{p}\}$.

For each of the semantics we have the usual relation between these two notions: observational equivalence implies testing equivalence.

Proposition 1. $\approx_o^s \subsetneq \approx_m^s$ *for* $s \in \{\mathsf{c}, \mathsf{e}, \mathsf{p}\}$.

Example 2. Consider processes A and B of Fig. 2. Process A computes a value $h^n(a)$ to be output on channel c, where $h^n(a)$ denotes n applications of h and $h^0(a) = a$. The value is initially a and A may choose to either output the current value, or update the current value by applying the free symbol h. B may choose non-deterministically to either behave as A or output the fresh name s. (The non-deterministic choice is encoded by a communication on the private channel e which may be received by either the process behaving as A or the process outputting s.)

We have that $A \not\approx_o^s B$. The two processes can indeed be distinguished by the context

$$\begin{aligned} C[\_] \mathrel{\hat{=}} \_ &\mid \mathsf{out}^{\mathsf{at}}(c_a,a) \mid !(\mathsf{in}^{\mathsf{at}}(c_a,x).\mathsf{out}^{\mathsf{at}}(c_a,h(x)) \\ &\mid \mathsf{in}^{\mathsf{at}}(c_a,y).\mathsf{in}^{\mathsf{at}}(c,z).\mathsf{if}\ y=z\ \mathsf{then}\ \mathsf{out}^{\mathsf{at}}(c_t,h(x)) \end{aligned}$$

Intuitively, when B outputs s the attacker context $C[\_]$ can iterate the application of h the same number of times as would have done process A. Comparing the value computed by the adversary ($h^n(a)$) and the honestly computed value (either $h^n(a)$ or s) the adversary distinguishes the two processes by outputting on the test channel c_t.

However, we have that $A \approx_m^s B$. Indeed, for any s-closing context $D[\_]$ and all public channel ch we have that $D[A] \Downarrow_{ch}^s$ if and only if $D[B] \Downarrow_{ch}^s$. In particular for context $C[\_]$ defined above we have that both $C[A] \Downarrow_{ch}^s$ and $C[B] \Downarrow_{ch}^s$ for $ch \in \{c_a, c_t, c\}$. Unlike observational equivalence, may testing does not require to "mimic" the other process stepwise and we cannot force a process into a particular branch.

2.4. Labelled semantics

The internal reduction semantics introduced in the previous section requires to reason about arbitrary contexts. Similar to the original applied pi calculus, we extend the three operational semantics by a *labeled operational semantics* which allows processes to directly interact with the (adversarial) environment: we define the relation $\xrightarrow{\ell}_{\mathsf{c}}$, $\xrightarrow{\ell}_{\mathsf{p}}$ and $\xrightarrow{\ell}_{\mathsf{e}}$ where ℓ is part of the alphabet $\mathcal{A} = \{\tau, out(c,d), eav(c,d), in(c,w), \nu k.out(c,k), \nu k.eav(c,k) \mid c,d \in \mathcal{C}h, k \in \mathcal{X} \cup \mathcal{C}h$ and w is a term of any sort$\}$. The labeled rules are given in Fig. 3.

Consider our alphabet of actions $\mathcal{A}$ defined above. Given $w \in \mathcal{A}^*$, $s \in \{\mathsf{c}, \mathsf{p}, \mathsf{e}\}$ and an extended process A, we say that $A \xrightarrow{w}_s A_n$ when $A \xrightarrow{\ell_1}_s A_1 \xrightarrow{\ell_2}_s A_2 \xrightarrow{\ell_3}_s \dots \xrightarrow{\ell_n}_s A_n$ for some extended processes $A_1, \dots, A_n$ and $w = \ell_1 \cdot \dots \cdot \ell_n$. By convention, we say that $A \xrightarrow{\epsilon}_s A$ where ϵ is the empty word. Given $\mathsf{tr} \in (\mathcal{A} \setminus \{\tau\})^*$, we say that $A \overset{\mathsf{tr}}{\Rightarrow}_s A'$ when there exists $w \in \mathcal{A}^*$ such that tr is the word w where we remove all τ actions and $A \xrightarrow{w}_s A'$.

$$\begin{array}{llll}
\textsc{In} & \mathsf{in}^{\mathsf{ho}}(c,y).P \xrightarrow{in(c,t)}_s P\{^t/_y\} & \textsc{Scope} & \dfrac{A \xrightarrow{\ell}_s A' \quad u \text{ does not occur in } \ell}{\nu u.A \xrightarrow{\ell}_s \nu u.A'} \\[2ex]
\textsc{Out-Ch} & \mathsf{out}^{\mathsf{ho}}(c,d).P \xrightarrow{out(c,d)}_s P & & \\[2ex]
\textsc{Open-Ch} & \dfrac{A \xrightarrow{out(c,d)}_s A' \qquad d \neq c}{\nu d.A \xrightarrow{\nu d.out(c,d)}_s A'} & \textsc{Par} & \dfrac{A \xrightarrow{\ell}_s A' \quad \begin{array}{c} bn(\ell)\cap fn(B)=\emptyset \\ bv(\ell)\cap fv(B)=\emptyset \end{array}}{A \mid B \xrightarrow{\ell}_s A' \mid B} \\[2ex]
\textsc{Eav-OCh} & \dfrac{A \xrightarrow{eav(c,d)}_{\mathsf{e}} A' \qquad d \neq c}{\nu d.A \xrightarrow{\nu d.eav(c,d)}_{\mathsf{e}} A'} & \textsc{Struct} & \dfrac{A \equiv B \quad B \xrightarrow{\ell}_s B' \quad B' \equiv A'}{A \xrightarrow{\ell}_s A'}
\end{array}$$

$$\begin{array}{ll}
\textsc{Eav-Ch} & \mathsf{out}^{\mathsf{ho}}(c,d).P \mid \mathsf{in}^{\mathsf{ho}}(c,x).Q \xrightarrow{eav(c,d)}_{\mathsf{e}} P \mid Q\{^d/x\} \\[2ex]
\textsc{Eav-T} & \mathsf{out}^{\mathsf{ho}}(c,t).P \mid \mathsf{in}^{\mathsf{ho}}(c,x).Q \xrightarrow{\nu y.eav(c,y)}_{\mathsf{e}} P \mid Q\{^t/x\} \mid \{^t/y\} \\[2ex]
\textsc{Out-T} & \mathsf{out}^{\mathsf{ho}}(c,t).P \xrightarrow{\nu x.out(c,x)}_s P \mid \{^t/_x\} \\
 & \qquad\qquad\qquad\qquad x \notin fv(P) \cup v(t)
\end{array}$$

where $s \in \{\mathsf{c}, \mathsf{p}, \mathsf{e}\}$.

Fig. 3. Labeled semantics.

Example 3. Coming back to Example 1, we saw that $A \xrightarrow{\tau}_{\mathsf{c}} \xrightarrow{\tau}_{\mathsf{c}} \nu d.(P\{^t/_x\} \mid Q\{^d/_y\})$ and no τ-actions in the other two semantics were available. Instead of explicitly adding eavesdrop actions, we can apply the rules Eav-OCh and Eav-T and obtain that

$$\begin{aligned}
A &\xrightarrow{\nu d.eav(c,d)}_{\mathsf{e}} \mathsf{in}^{\mathsf{ho}}(d,x).P \mid \mathsf{out}^{\mathsf{ho}}(d,t).Q\{^d/_y\}) \\
&\xrightarrow{\nu z.eav(d,z)}_{\mathsf{e}} P\{^t/_x\} \mid Q\{^d/_y\} \mid \{^t/_z\}
\end{aligned}$$

We can now define both reachability and different equivalence properties in terms of these labelled semantics and relate them to the internal reduction. To define reachability properties in the labelled semantics, we define $A \downarrow\downarrow_c^s$ to hold when $A \overset{\mathsf{tr}}{\Rightarrow} A'$, $\mathsf{tr} = \mathsf{tr}_1 out(c,t)\mathsf{tr}_2$ and tr_1 does not bind c for some $\mathsf{tr}, \mathsf{tr}_1, \mathsf{tr}_2 \in (\mathcal{A} \setminus \{\tau\})^*$, term t and extended process A'.

The following proposition states that any reachability property modelled in terms of $A \Downarrow_c^{s,\theta}$ and universal quantification over processes, can also be expressed using $A \downarrow\downarrow_c^s$ without the need to quantify over processes.

Proposition 2. *For all closed honest plain processes A, for all $s \in \{\mathsf{c}, \mathsf{e}, \mathsf{p}\}$, $A \downarrow\downarrow_c^s$ iff there exists an attacker plain process I^s such that $I^s \mid A \Downarrow_c^{s,\mathsf{ho}}$.*

Next, we define equivalence relations using our labelled semantics that may serve as proof techniques for the may testing relation. First we need to define an indistinguishability relation on frames, called static equivalence [1].

Definition 3 (Static equivalence $\sim$)**.** Two terms u and v are *equal in the frame* ϕ, written $(u =_{\mathsf{E}} v)\phi$, if there exists $\tilde{n}$ and a substitution σ such that $\phi \equiv \nu\tilde{n}.\sigma$, $\tilde{n} \cap (fn(u) \cup fn(v)) = \emptyset$, and $u\sigma =_{\mathsf{E}} v\sigma$.

Two closed frames ϕ_1 and ϕ_2 are *statically equivalent*, written $\phi_1 \sim \phi_2$, when:

- $\mathrm{dom}(\phi_1) = \mathrm{dom}(\phi_2)$, and
- for all terms u, v we have that: $(u =_{\mathsf{E}} v)\phi_1$ if and only if $(u =_{\mathsf{E}} v)\phi_2$.

Example 4. Consider the equational theory generated by the equation $\mathsf{dec}(\mathsf{enc}(x, y), y) = x$. Then we have that

$$\begin{aligned} \nu k.\, \{^{\mathsf{enc}(a,k)}/_{x_1}\} &\sim \nu k.\, \{^{\mathsf{enc}(b,k)}/_{x_1}\} \\ \nu k.\, \{^{\mathsf{enc}(a,k)}/_{x_1}, {}^{k}/_{x_2}\} &\not\sim \nu k.\, \{^{\mathsf{enc}(b,k)}/_{x_1}, {}^{k}/_{x_2}\} \\ \nu k, a.\, \{^{\mathsf{enc}(a,k)}/_{x_1}, {}^{k}/_{x_2}\} &\sim \nu k, b.\, \{^{\mathsf{enc}(b,k)}/_{x_1}, {}^{k}/_{x_2}\} \end{aligned}$$

Intutively, the first equivalence confirms that encryption hides the plaintext when the decryption key is unknown. The second equivalence does not hold as the test $(\mathsf{dec}(x_1, x_2) =_{\mathsf{E}} a)$ holds on the left hand side, but not on the right hand side. Finally, the third equivalence again holds as two restricted names are indistinguishable.

Now we are ready to define two classical equivalences on processes, based on the labelled semantics: trace equivalence and labelled bisimulation.

Definition 4 (Trace equivalences $\approx_t^{\mathsf{c}}$, $\approx_t^{\mathsf{p}}$, $\approx_t^{\mathsf{e}}$)**.** Let $s \in \{\mathsf{c}, \mathsf{p}, \mathsf{e}\}$. Let A and B be two closed honest extended processes. We say that $A \sqsubseteq_t^s B$ if for all $A \overset{\mathsf{tr}}{\Rightarrow}_s A'$ such that $bn(\mathsf{tr}) \cap fn(B) = \emptyset$, there exists B' such that $B \overset{\mathsf{tr}}{\Rightarrow}_s B'$ and $\phi(A') \sim \phi(B')$. We say that $A \approx_t^s B$ when $A \sqsubseteq_t^s B$ and $B \sqsubseteq_t^s A$.

Definition 5 (Labeled bisimulations $\approx_\ell^{\mathsf{c}}$, $\approx_\ell^{\mathsf{p}}$, $\approx_\ell^{\mathsf{e}}$)**.** Let $s \in \{\mathsf{c}, \mathsf{p}, \mathsf{e}\}$. Let A and B two closed honest extended processes such that $\mathrm{dom}(A) = \mathrm{dom}(B)$. We say that $A \approx_\ell^s B$ if $\approx_\ell^s$ is the largest equivalence relation such that:

- $\phi(A) \sim \phi(B)$
- $A \xrightarrow{\tau}_s A'$ implies $B \overset{\epsilon}{\Rightarrow}_s B'$ and $A' \approx_\ell^s B'$ for some B',
- $A \xrightarrow{\ell}_s A'$ and $bn(\ell) \cap fn(B) = \emptyset$ implies $B \overset{\ell}{\Rightarrow}_s B'$ and $A' \approx_\ell^s B'$ for some B'.

We again have, as usual that labelled bisimulation implies trace equivalence.

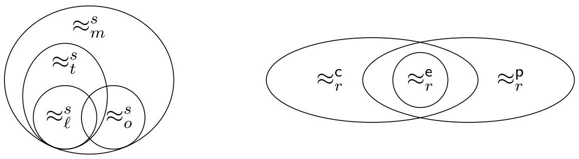
Proposition 3. *$\approx_\ell^s \subsetneq \approx_t^s$ for $s \in \{\mathsf{c}, \mathsf{e}, \mathsf{p}\}$.*

In [1] it is shown that $\approx_o^{\mathsf{c}} = \approx_\ell^{\mathsf{c}}$. We conjecture that for the new semantics p and e this same equivalence holds as well. Re-showing these results is beyond the scope of this paper, and we will mainly focus on testing/trace equivalence. As shown in [15], for the classical semantics trace equivalence implies may testing, while the converse does not hold in general. The two relations do however coincide on image-finite processes.

Definition 6. Let A be a closed extended process. A is *image-finite* for the semantics $s \in \{\mathsf{c}, \mathsf{e}, \mathsf{p}\}$ if for each trace tr the set of equivalence classes $\{\phi(B) \mid A \overset{\mathsf{tr}}{\Rightarrow}_s B\}/\!\sim$ is finite.

Note that any replication-free process is necessarily image-finite as there are only a finite number of possible traces for any given sequence of labels tr. The same relations among trace equivalence and may testing shown for the classical semantics hold also for the other semantics.

Theorem 1. *$\approx_t^s \subsetneq \approx_m^s$ and $\approx_t^s = \approx_m^s$ on image-finite processes for $s \in \{\mathsf{c}, \mathsf{e}, \mathsf{p}\}$.*



for all $s \in \{\mathsf{c}, \mathsf{p}, \mathsf{e}\}$
for image finite processes $\approx_t^s = \approx_m^s$
if $s = \mathsf{c}$ then $\approx_\ell^s = \approx_o^s$ (conjectured for $s \in \{\mathsf{p}, \mathsf{e}\}$)

for all $r \in \{m, t, \ell\}$

Fig. 4. Overview of the results.

The proof of this result (for the classical semantics) is given in [15] and is easily adapted to the other semantics. To see that the implication is strict, we continue Example 2 on processes A and B defined in Fig. 2. We already noted that $A \approx_m^s B$, but will now show that $A \not\approx_t^s B$ (for $s \in \{\mathsf{c}, \mathsf{e}, \mathsf{p}\}$). All possible traces of A are of the form $A \xRightarrow{\nu x.out(c,x)}_s A'$ where $\phi(A') = \{^{h^n(a)}/x\}$ for $n \in \mathbb{N}$. We easily see that $A \not\approx_t^s B$ as for any n we have that $\{^{h^n(a)}/_x\} \not\sim \{^s/_x\}$, by testing $x = h^n(a)$. On the other hand, given an image-finite process, we can only have a finite number of different frames for a given trace, and therefore we can bound the context size that is necessary for distinguishing the processes.

3. Comparing the different semantics

In this section we state our results on comparing these semantics. Results on equivalence comparison are summarized in Fig. 4.

We first show that, as expected, all the semantics coincide for reachability properties.

Theorem 2. *For all ground, closed honest extended processes A, for all channels d, we have that $A \Downarrow_d^{\mathsf{p}}$ iff $A \Downarrow_d^{\mathsf{c}}$ iff $A \Downarrow_d^{\mathsf{e}}$.*

The next result is, in our opinion, more surprising. As the private semantics force the adversary to observe all information, one might expect that his distinguishing power increases over the classical one. This intuition is however wrong: the classical and private trace equivalences, testing equivalence and labelled bisimulations appear to be incomparable.

Theorem 3. *$\approx_r^{\mathsf{p}} \not\subseteq \approx_r^{\mathsf{c}}$ and $\approx_r^{\mathsf{c}} \not\subseteq \approx_r^{\mathsf{p}}$ for $r \in \{\ell, t, m\}$.*

Proof. We show both statements separately.

$\approx_r^{\mathsf{p}} \not\subseteq \approx_r^{\mathsf{c}}$. We first show that there exist A and B such that $A \approx_\ell^{\mathsf{p}} B$, but $A \not\approx_m^{\mathsf{c}} B$. Note that, as $\approx_\ell^s \subset \approx_t^s \subseteq \approx_m^s$ for $s \in \{\mathsf{c}, \mathsf{p}\}$ these processes demonstrate both that $\approx_\ell^{\mathsf{p}} \not\subseteq \approx_\ell^{\mathsf{c}}$, $\approx_t^{\mathsf{p}} \not\subseteq \approx_t^{\mathsf{c}}$ and $\approx_m^{\mathsf{p}} \not\subseteq \approx_m^{\mathsf{c}}$.

Consider processes A and B defined in Fig. 5. In short, the result follows from the fact that if A performs an internal communication on channel c followed by an output on d (from P_1), B has no choice other then performing the output on d in P_2. In the private semantics, however, the internal communication will be split in an output followed by an input: after the output on c, the input $\mathsf{in}^{\mathsf{ho}}(c, x).P_2(x)$ following the output becomes available. More precisely, to see that $A \approx_\ell^{\mathsf{p}} B$ we first observe that if $A \xrightarrow{\nu z.out(c,z)}_{\mathsf{p}} A'$ then $B \xrightarrow{\nu z.out(c,z)}_{\mathsf{p}} B'$ and $A' \equiv B'$, and vice-versa. If $A \xrightarrow{in(c,t)}_{\mathsf{p}} A'$ then $B \xrightarrow{in(c,t)}_{\mathsf{p}} B'$.

$$A \mathrel{\hat{=}} \nu s_1.\nu s_2.((\mathsf{out}^{\mathsf{ho}}(c, s_1).\mathsf{in}^{\mathsf{ho}}(c, x).P_1(x)) \mid (\mathsf{in}^{\mathsf{ho}}(c, y).P_2(y)))$$
$$B \mathrel{\hat{=}} \nu s_1.\nu s_2.((\mathsf{out}^{\mathsf{ho}}(c, s_1).\mathsf{in}^{\mathsf{ho}}(c, x).P_2(x)) \mid (\mathsf{in}^{\mathsf{ho}}(c, y).P_1(y)))$$

where

$$P_1(x) \mathrel{\hat{=}} (\mathsf{if}\ x = s_1\ \mathsf{then}\ \mathsf{out}^{\mathsf{ho}}(d, s_2)) \mid (\mathsf{if}\ x = s_2\ \mathsf{then}\ \mathsf{out}^{\mathsf{ho}}(e, x))$$
$$P_2(x) \mathrel{\hat{=}} (\mathsf{if}\ x = s_1\ \mathsf{then}\ \mathsf{out}^{\mathsf{ho}}(d, s_2))$$

To emit on channel e, processes A and B must execute $P_2(s_1)$ followed by $P_1(s_2)$. In the classical semantics, a trace of A emitting on e through an internal communication between $\mathsf{out}^{\mathsf{ho}}(c, s_1)$ and $\mathsf{in}^{\mathsf{ho}}(c, y)$ forces B to execute $P_1(s_1)$ thus preventing it to emit on e.

Fig. 5. Processes A and B such that $A \approx_\ell^{\mathsf{p}} B$ and $A \not\approx_m^{\mathsf{c}} B$.

As $t \notin \{s_1, s_2\}$ we have that $P_1(t) \approx_\ell^{\mathsf{p}} 0 \approx_\ell^{\mathsf{p}} P_2(t)$. Finally, if $t \neq s_2$ we also have that $P_1(t) \approx_\ell^{\mathsf{p}} P_2(t)$ as in particular $P_1(s_1) \approx_\ell^{\mathsf{p}} P_2(s_1)$. Therefore,

$$\nu s_1.\nu s_2.(\mathsf{out}^{\mathsf{ho}}(c, s_1).\mathsf{in}^{\mathsf{ho}}(c, x).P_1(x)) \approx_\ell^{\mathsf{p}} \nu s_1.\nu s_2.(\mathsf{out}^{\mathsf{ho}}(c, s_1).\mathsf{in}^{\mathsf{ho}}(c, x).P_2(x))$$

which allows us to conclude.

As A and B are image-finite, we have that $A \approx_m^{\mathsf{c}} B$ if and only if $A \approx_t^{\mathsf{c}} B$. To see that $A \not\approx_t^{\mathsf{c}} B$ we observe that A may perform the following transition sequence, starting with an internal communication on a public channel:

$$\begin{aligned} A &\xrightarrow{\tau}_{\mathsf{c}} \nu s_1.\nu s_2.((\mathsf{in}^{\mathsf{ho}}(c, x).P_1(x)) \mid (P_2(s_1))) \\ &\xRightarrow{\nu z.out(d,z)}_{\mathsf{c}} \nu s_1.\nu s_2.((\mathsf{in}^{\mathsf{ho}}(c, x).P_1(x)) \mid \{^{s_2}/_z\}) \\ &\xrightarrow{in(c,z)}_{\mathsf{c}} \nu s_1.\nu s_2.(P_1(s_2) \mid \{^{s_2}/_z\}) \end{aligned}$$

In order to mimic the behaviour of A, B must perform the same sequence of observable transitions:

$$B \xRightarrow{\nu z.out(d,z)\ in(c,z)}_{\mathsf{c}} \nu s_1.\nu s_2.(P_2(s_2) \mid \{^{s_2}/_z\})$$

We conclude as $\nu s_1.\nu s_2.(P_1(s_2) \mid \{^{s_2}/_z\}) \xrightarrow{\nu z'.out(e,z')} \nu s_1.\nu s_2.(\{^{s_2}/_z\} \mid \{^{s_2}/_{z'}\})$, but $\nu s_1.\nu s_2.(P_2(s_2) \mid \{^{s_2}/_z\}) \not\xrightarrow{\nu z'.out(e,z')}$. This trace inequivalence has also been shown using DeepSec.

$\approx_r^{\mathsf{c}} \not\subseteq \approx_r^{\mathsf{p}}$. To show that $\approx_r^{\mathsf{c}} \not\subseteq \approx_r^{\mathsf{p}}$ for $r \in \{\ell, t, m\}$ we show that there exist processes A and B such that $A \approx_\ell^{\mathsf{c}} B$ and $A \not\approx_m^{\mathsf{p}} B$. As in the first part of the proof, note that, as $\approx_\ell^s \subset \approx_t^s \subseteq \approx_m^s$ for $s \in \{\mathsf{c}, \mathsf{p}\}$ these processes demonstrate that $\approx_\ell^{\mathsf{c}} \not\subseteq \approx_\ell^{\mathsf{p}}$, $\approx_t^{\mathsf{c}} \not\subseteq \approx_t^{\mathsf{p}}$ and $\approx_m^{\mathsf{c}} \not\subseteq \approx_m^{\mathsf{p}}$.

Consider the processes A and B defined in Fig. 6. The proof crucially relies on the fact that B may perform an internal communication in the classical semantics to mimic A, which becomes visible in the attacker in the private semantics. To see that $A \approx_\ell^{\mathsf{c}} B$ we first observe that the only first possible action from A or B is an input. In particular, given a term t, there is a unique B' such that $B \xrightarrow{in(c,t)} B'$ where $B' = \nu s.(\mathsf{out}^{\mathsf{ho}}(c, s).\mathsf{out}^{\mathsf{ho}}(d, a) \mid \mathsf{in}^{\mathsf{ho}}(c, y).P(y))$. However, if $A \xrightarrow{in(c,t)} A'$ then either $A' = B'$ or $A' = A''$ with $A'' \mathrel{\hat{=}} \nu s.(\mathsf{in}^{\mathsf{ho}}(c, x).\mathsf{out}^{\mathsf{ho}}(c, s).\mathsf{out}^{\mathsf{ho}}(d, a) \mid P(t))$. Therefore, to complete the proof, we

$$A \mathrel{\hat{=}} \nu s.(\mathsf{in}^{\mathsf{ho}}(c,x).\mathsf{out}^{\mathsf{ho}}(c,s).\mathsf{out}^{\mathsf{ho}}(d,a) \mid \mathsf{in}^{\mathsf{ho}}(c,y).P(y))$$
$$B \mathrel{\hat{=}} \nu s.(\mathsf{in}^{\mathsf{ho}}(c,x).(\mathsf{out}^{\mathsf{ho}}(c,s).\mathsf{out}^{\mathsf{ho}}(d,a) \mid \mathsf{in}^{\mathsf{ho}}(c,y).P(y)))$$

where

$$P(y) \mathrel{\hat{=}} \mathsf{if}\ y = s\ \mathsf{then}\ \mathsf{in}^{\mathsf{ho}}(c,z).\mathsf{out}^{\mathsf{ho}}(c,s).\mathsf{out}^{\mathsf{ho}}(d,a)\ \mathsf{else}\ \mathsf{out}^{\mathsf{ho}}(d,a)$$

In the private semantics, a trace of A starting with the execution of $\mathsf{in}^{\mathsf{ho}}(c,y)$ can only be matched on B by executing $\mathsf{in}^{\mathsf{ho}}(c,x)$. B could then emit on channel c, which is not the case for A, hence yielding non equivalence. In the classic semantics, an internal communication between $\mathsf{out}^{\mathsf{ho}}(c,s)$ and $\mathsf{in}^{\mathsf{ho}}(c,y)$ allows to *hide* the fact that B can emit on c.

Fig. 6. Processes A and B such that $A \approx^{\mathsf{c}}_{\ell} B$ and $A \not\approx^{\mathsf{p}}_{m} B$.

only need to find B'' such that $B \overset{in(c,t)}{\Longrightarrow} B''$ and $A'' \approx^{\mathsf{c}}_{\ell} B''$. Such a process can be obtained by applying an internal communication on B', *i.e.* $B \xrightarrow{in(c,t)}_{\mathsf{c}} B' \xrightarrow{\tau} \nu s.(\mathsf{out}^{\mathsf{ho}}(d,a) \mid P(s))$. Note that $t \neq s$ since s is bound, meaning that $P(t) \approx^{\mathsf{c}}_{\ell} \mathsf{out}^{\mathsf{ho}}(d,a)$. Moreover, $P(s) \approx^{\mathsf{c}}_{\ell} \mathsf{in}^{\mathsf{ho}}(c,x).\mathsf{out}^{\mathsf{ho}}(c,s).\mathsf{out}^{\mathsf{ho}}(d,a)$. This allows us to conlude that $\nu s.(\mathsf{out}^{\mathsf{ho}}(d,a) \mid P(s)) \approx^{\mathsf{c}}_{\ell} A''$.

Again, as A and B are image-finite may and trace equivalence coincide. To see that $A \not\approx^{\mathsf{p}}_{t} B$ we first observe that A may perform the following transition sequence:

$$\begin{aligned} A \xrightarrow{in(c,t)}_{\mathsf{p}} A'' \xrightarrow{\tau}_{\mathsf{p}}\ & \nu s.(\mathsf{in}^{\mathsf{ho}}(c,x).\mathsf{out}^{\mathsf{ho}}(c,s).\mathsf{out}^{\mathsf{ho}}(d,a) \mid \mathsf{out}^{\mathsf{ho}}(d,a)) \\ \xrightarrow{\nu z.out(d,z)}_{\mathsf{p}}\ & \nu s.(\mathsf{in}^{\mathsf{ho}}(c,x).\mathsf{out}^{\mathsf{ho}}(c,s).\mathsf{out}^{\mathsf{ho}}(d,a) \mid \{^{a}/_{z}\}) \end{aligned}$$

We conclude as $B \xrightarrow{in(c,t)}_{\mathsf{p}} B'$ but $B' \overset{\nu z.out(d,z)}{\not\longrightarrow}_{\mathsf{p}}$. Violation of this trace equivalence has also been shown using the DeepSec tool. □

One may also note that the counter-example witnessing that equivalences in the private semantics do not imply equivalences in the classical semantics is *minimal*: it does not use function symbols, equational reasoning, private channels, replication nor else branches. The second part of the proof relies on the use of else branches. We can however refine this result in the case of labeled bisimulation to processes without else branches, the counter-example being the same processes A and B described in the proof but where we replace each $\mathsf{out}^{\mathsf{ho}}(d,a)$ by 0. In the case of trace equivalence, we can also produce a counter-example without else branches witnessing that trace equivalences in the classical semantics do no imply trace equivalences in the private semantics but provided that we rely on a function symbol h. In the Appendix, we describe in more details these processes and give the proofs of them being counter-examples.

Next, we show that the eavesdropping semantics yields strictly stronger bisimulations, trace and may testing equivalences: the eavesdropping semantics is actually strictly included in the intersection of the classic and private semantics.

Theorem 4. $\approx^{\mathsf{e}}_{t} \subsetneq \approx^{\mathsf{p}}_{t} \cap \approx^{\mathsf{c}}_{t}$.

Proof sketch. We show the result in 3 steps: we show that (1) $\approx^{\mathsf{e}}_{t} \subseteq \approx^{\mathsf{p}}_{t}$, (2) $\approx^{\mathsf{e}}_{t} \subseteq \approx^{\mathsf{c}}_{t}$, and (3) that the implication is strict, i.e., there exist A, B such that $A \approx^{\mathsf{p}}_{t} B$, $A \approx^{\mathsf{c}}_{t} B$ and $A \not\approx^{\mathsf{e}}_{t} B$.

$$A \mathrel{\hat{=}} \nu s_1.\nu s_2.((\mathsf{out}^{\mathsf{ho}}(c, s_1).\mathsf{in}^{\mathsf{ho}}(c, x).P_1(x)) \mid (\mathsf{in}^{\mathsf{ho}}(c, y).P_2(y)))$$
$$B \mathrel{\hat{=}} \nu s_1.\nu s_2.((\mathsf{out}^{\mathsf{ho}}(c, s_1).\mathsf{in}^{\mathsf{ho}}(c, x).P_2(x)) \mid (\mathsf{in}^{\mathsf{ho}}(c, y).P_1(y)))$$

where

$$P_1(x) \mathrel{\hat{=}} (\mathsf{if}\ x = s_1\ \mathsf{then}\ \mathsf{in}^{\mathsf{ho}}(d, z).\mathsf{if}\ z = s_1\ \mathsf{then}\ \mathsf{out}^{\mathsf{ho}}(d, s_2)) \mid (\mathsf{if}\ x = s_2\ \mathsf{then}\ \mathsf{out}^{\mathsf{ho}}(e, x))$$
$$P_2(x) \mathrel{\hat{=}} (\mathsf{if}\ x = s_1\ \mathsf{then}\ \mathsf{in}^{\mathsf{ho}}(d, z).\mathsf{if}\ z = s_1\ \mathsf{then}\ \mathsf{out}^{\mathsf{ho}}(d, s_2))$$

To emit on channel e, processes A and B must execute $P_2(s_1)$ by inputing twice s_1 followed by $P_1(s_2)$. In the classical semantics, an internal communication on A between $\mathsf{out}^{\mathsf{ho}}(c, s_1)$ and $\mathsf{in}^{\mathsf{ho}}(c, y)$ forces B to execute $P_1(s_1)$ but *hides* s_1, preventing a second input of s_1 by A. However, in the eavesdropping semantics, the internal communication *reveals* s_1 allowing A to emit on e but not B.

Fig. 7. Processes A and B such that $A \approx_\ell^{\mathsf{c}} B$, $A \approx_\ell^{\mathsf{p}} B$ but $A \not\approx_t^{\mathsf{e}} B$.

(1) We first prove that $\approx_t^{\mathsf{e}} \subseteq \approx_t^{\mathsf{p}}$. Suppose that $A \approx_t^{\mathsf{e}} B$. We need to show that for any A' such that $A \overset{tr}{\Rightarrow}_{\mathsf{p}} A'$ there exists B' such that $B \overset{tr}{\Rightarrow}_{\mathsf{p}} B'$. It follows from the definition of the semantics that whenever $A \overset{\mathsf{tr}}{\Rightarrow}_{\mathsf{p}} A'$ then we also have $A \overset{\mathsf{tr}}{\Rightarrow}_{\mathsf{e}} A'$ as $\overset{\ell}{\rightarrow}_{\mathsf{p}} \subset \overset{\ell}{\rightarrow}_{\mathsf{e}}$. As $A \approx_t^{\mathsf{e}} B$, we have that there exists B', such that $B \overset{\mathsf{tr}}{\Rightarrow}_{\mathsf{e}} B'$ and $\phi(A') \sim \phi(B')$. As tr does not contain labels of the form $eav(c, d)$ nor $\nu y.eav(c, y)$ and as no COMM-EAV are possible (A and B are honest processes) we also have that $B \overset{\mathsf{tr}}{\Rightarrow}_{\mathsf{p}} B'$. Hence $A \approx_t^{\mathsf{p}} B$.

(2) We next prove that $\approx_t^{\mathsf{e}} \subseteq \approx_t^{\mathsf{c}}$. Similar to Item 1 we suppose that $A \approx_t^{\mathsf{e}} B$ and $A \overset{tr_{\mathsf{c}}}{\Rightarrow}_{\mathsf{c}} A'_{\mathsf{c}}$. From the semantics, we obtain that $A \overset{tr_{\mathsf{e}}}{\Rightarrow}_{\mathsf{e}} A'_{\mathsf{e}}$, where

- $\phi(A'_{\mathsf{c}}) \subseteq \phi(A'_{\mathsf{e}})$, i.e., $\mathrm{dom}(\phi(A'_{\mathsf{c}})) \subseteq \mathrm{dom}(\phi(A'_{\mathsf{e}}))$ and the frames coincide on the common domain.
- tr_{e} is constructed from tr_{c} by replacing any τ action resulting from the COMM rule by an application of an eavesdrop rule (EAV-T, EAV-CH, or EAV-OCH).

The proof is done by induction on the length of tr_{c} and the proof tree of each transition. As $A \approx_t^{\mathsf{e}} B$ we also have that $B \overset{tr_{\mathsf{e}}}{\Rightarrow}_{\mathsf{e}} B'_{\mathsf{e}}$ and $A'_{\mathsf{e}} \sim B'_{\mathsf{e}}$. We show by the definition of the semantics that $B \overset{tr_{\mathsf{c}}}{\Rightarrow}_{\mathsf{c}} B'_{\mathsf{c}}$ and $\phi(B'_{\mathsf{c}}) \subseteq \phi(B'_{\mathsf{e}})$ (replacing each eavesdrop action by an internal communication). Due to the inclusions of the frames and $A'_{\mathsf{e}} \sim B'_{\mathsf{e}}$ we also have that $A'_{\mathsf{c}} \sim B'_{\mathsf{c}}$.

(3) Finally we show that the implication $\approx_t^{\mathsf{e}} \subsetneq \approx_t^{\mathsf{p}} \cap \approx_t^{\mathsf{c}}$ is strict, i.e., there exist A and B such that $A \approx_\ell^{\mathsf{c}} B$ (which implies $A \approx_t^{\mathsf{c}} B$), $A \approx_\ell^{\mathsf{p}} B$ (which implies $A \approx_t^{\mathsf{p}} B$) but $A \not\approx_t^{\mathsf{e}} B$.
Consider the processes A and B defined in Fig. 7. This example is a variant of the one given in Fig. 5. The difference is the addition of "$\mathsf{in}^{\mathsf{ho}}(d, z).\mathsf{if}\ z = s_1\ \mathsf{then}$ " in processes $P_1(x)$ and $P_2(x)$: this additional check is used to verify whether the adversary learned s_1 or not. The proofs that $A \approx_\ell^{\mathsf{c}} B$ and $A \approx_\ell^{\mathsf{p}} B$ follow the same lines as in Theorem 3. We just additionally observe that $\nu s_1.(\mathsf{in}^{\mathsf{ho}}(d, z).\mathsf{if}\ z = s_1\ \mathsf{then}\ \mathsf{out}^{\mathsf{ho}}(d, s_2)) \approx_\ell^{s} \nu s_1.\ (\mathsf{in}^{\mathsf{ho}}(d, z).0)$ for $s \in \{\mathsf{c}, \mathsf{p}\}$.
The trace witnessing that $A \not\approx_t^{\mathsf{e}} B$ is again similar to the one in Theorem 3, but starting with an eavesdrop transition which allows the attacker to learn s_1, which in turn allows him to learn s_2 and distinguish $P_1(s_2)$ from $P_2(s_2)$. These trace (in)equivalences have also been verified using DeepSec. □

We note from the processes defined in Fig. 7 that the implications are strict even for processes that do not communicate on private channels, do not use replication, nor else branches and terms are simply names (no function symbols nor equational theories).

Theorem 5. $\approx_\ell^{\mathsf{e}} \subsetneq \approx_\ell^{\mathsf{p}} \cap \approx_\ell^{\mathsf{c}}$.

Proof sketch. The proof is structured in 3 steps, as in the proof of Theorem 4.

(1) We first show that $\approx_\ell^{\mathsf{e}} \subseteq \approx_\ell^{\mathsf{p}}$. Suppose $A \approx_\ell^{\mathsf{e}} B$ and let $\mathcal{R}$ be the relation witnessing this equivalence. We will show that $\mathcal{R}$ is also a labelled bisimulation in the private semantics. Suppose $A \mathrel{\mathcal{R}} B$.

- as $A \approx_\ell^{\mathsf{e}} B$, we have that $\phi(A) \sim \phi(B)$.
- if $A \xrightarrow{\tau}_{\mathsf{p}} A'$ then, as $\xrightarrow{\tau}_{\mathsf{p}} \subset \xrightarrow{\tau}_{\mathsf{e}}$, $A \xrightarrow{\tau}_{\mathsf{e}} A'$. As $A \approx_\ell^{\mathsf{e}} B$ there exists B' such that $B \overset{\epsilon}{\Rightarrow}_{\mathsf{e}} B'$ and $A' \mathrel{\mathcal{R}} B'$. As B is a honest process no COMM-EAV transition is possible, and hence $B \overset{\epsilon}{\Rightarrow}_{\mathsf{p}} B'$.
- if $A \xrightarrow{\ell}_{\mathsf{p}} A'$ and $bn(\ell) \cap fn(B) = \emptyset$ then we also have that $A \xrightarrow{\ell}_{\mathsf{e}} A'$ (as $\xrightarrow{\ell}_{\mathsf{p}} \subset \xrightarrow{\ell}_{\mathsf{e}}$ and there exists B' such that $B \overset{\ell}{\Rightarrow}_{\mathsf{e}} B'$ and $A' \mathrel{\mathcal{R}} B'$. As no COMM-EAV are possible and ℓ is not of the form $eav(c, d)$ nor $\nu y.eav(c, y)$ we have that $B \overset{\ell}{\Rightarrow}_{\mathsf{p}} B'$.

(2) We next show that $\approx_\ell^{\mathsf{e}} \subseteq \approx_\ell^{\mathsf{c}}$. We will show that $\approx_\ell^{\mathsf{e}}$ is also a labelled bisimulation in the classical semantics. The proof relies on similar arguments as in Item 2 of the proof of Theorem 4 and the facts that

- $\nu\tilde{n}.(A' \mid \{^t/_x\}) \approx_\ell^{\mathsf{e}} \nu\tilde{n}.(B' \mid \{^u/_x\})$ implies $\nu\tilde{n}.A' \approx_\ell^{\mathsf{e}} \nu\tilde{n}.B'$,
- $A' \approx_\ell^{\mathsf{e}} B'$ implies $\nu c.A' \approx_\ell^{\mathsf{e}} \nu c.B'$

The first property is needed when an internal communication of a term or public channel is replaced by an eavesdrop action and an input. The second property handles the case when we replace the internal communication of a private channel by an application of the EAV-OCH rule and an input.

(3) To show that the implication $\approx_\ell^{\mathsf{e}} \subsetneq \approx_\ell^{\mathsf{p}} \cap \approx_\ell^{\mathsf{c}}$ is strict, we exhibit processes A and B such that $A \approx_\ell^{\mathsf{c}} B$, $A \approx_\ell^{\mathsf{p}} B$ but $A \not\approx_t^{\mathsf{e}} B$ (which implies $A \not\approx_\ell^{\mathsf{e}} B$). The processes defined in Fig. 7 witness this fact (cf the discussion of these processes in the proof of Theorem 4). □

Again we note that the implications are strict, even for processes containing only public channels.

Theorem 6. $\approx_m^{\mathsf{e}} \subsetneq \approx_m^{\mathsf{p}} \cap \approx_m^{\mathsf{c}}$.

Proof sketch. The proof is structured in 3 parts, as for Theorems 5 and 4.

(1) We first prove that $\approx_m^{\mathsf{e}} \subseteq \approx_m^{\mathsf{p}}$. Suppose that $A \approx_m^{\mathsf{e}} B$. Suppose that $A \approx_m^{\mathsf{e}} B$. We need to show that for all channel c, for all $C[\_]$ attacker evaluation contexts p-closing for A and B, $C[A] \Downarrow_c^{\mathsf{p}}$ is equivalent to $C[B] \Downarrow_c^{\mathsf{p}}$. It follows from the definition of the private semantics that any process $\mathsf{eav}(c, x).P$ in $C[\_]$ has the same behaviour as the process 0. Hence, we generate a context $C^1[\_]$ by replacing in $C[\_]$ any instance of $\mathsf{eav}(c, x).P$ by 0, and thus obtaining $C[A] \Downarrow_c^{\mathsf{p}} \Leftrightarrow C'[A] \Downarrow_c^{\mathsf{p}}$ and $C[B] \Downarrow_c^{\mathsf{p}} \Leftrightarrow C'[B] \Downarrow_c^{\mathsf{p}}$. Notice that the definition of semantics gives us $\rightarrow_{\mathsf{p}} \subseteq \rightarrow_{\mathsf{e}}$. Hence, $C'[A] \Downarrow_c^{\mathsf{p}}$ implies $C'[A] \Downarrow_c^{\mathsf{e}}$ and $C'[B] \Downarrow_c^{\mathsf{p}}$ implies $C'[B] \Downarrow_c^{\mathsf{e}}$. Furthermore, since we built $C'[\_]$ to not contain any process of the form $\mathsf{eav}(c, x).P$, we deduce that rules C-EAV and C-OEAV can never be applied in a derivation of $C'[A]$ or $C'[B]$. It implies that $C'[A] \Downarrow_c^{\mathsf{p}} \Leftrightarrow C'[A] \Downarrow_c^{\mathsf{e}}$ and

$C'[B] \Downarrow^{\mathsf{p}}_c \Leftrightarrow C'[B] \Downarrow^{\mathsf{e}}_c$. Thanks to $A \approx^{\mathsf{e}}_m B$, we know that $C'[A] \Downarrow^{\mathsf{e}}_c \Leftrightarrow C'[B] \Downarrow^{\mathsf{e}}_c$ and so we conclude that $C[A] \Downarrow^{\mathsf{p}}_c \Leftrightarrow C[B] \Downarrow^{\mathsf{p}}_c$.

(2) We next prove that $\approx^{\mathsf{e}}_m \subseteq \approx^{\mathsf{c}}_m$. Similarly to Item 1, we consider a channel c and an attacker evaluation context $C[\_]$ that is c-closing for A and B. The main difficulty of this proof is to match the application of the rule Comm in the classical semantics with the rules C-Eav and C-OEac. However, $C[\_]$ does not necessarily contain eavesdrop process $\mathsf{eav}(d, x) \mid \omega c$. Moreover, as mentioned in Item 1, a process $\mathsf{eav}(d, x).P$ has the same behavior as 0 in the classical semantics but can have a completely different behaviour in the eavesdropping semantics if P is not 0. Thus, we remove from $C[\_]$ the eavesdrop processes, obtaining $C'[\_]$. Then, we define a new context $C''[\_]$ based on $C'[\_]$ where will add harmless eavesdrop process $\mathsf{eav}(d, y).0$. We first add in parallel the processes $!\mathsf{eav}(a, y) \mid \omega a$ for all free channels a in $C'[\_]$, A and B. Moreover, since private channels can be opened, we also replace any process $\nu d.P$, $\mathsf{in}^{\mathsf{at}}(c, x).P$ where d, x are of channel type with $\nu d.(P \ |!\mathsf{eav}(d, y))$ and $\mathsf{in}^{\mathsf{at}}(c, x).(P \ |!\mathsf{eav}(x, y))$. By induction of the derivations, we can show that $C[A] \Downarrow^{\mathsf{c}}_c \Leftrightarrow C''[A] \Downarrow^{\mathsf{e}}_c$ and $C[B] \Downarrow^{\mathsf{c}}_c \Leftrightarrow C''[B] \Downarrow^{\mathsf{e}}_c$. Since $A \approx^{\mathsf{e}}_m B$, we deduce that $C''[A] \Downarrow^{\mathsf{e}}_c \Leftrightarrow C''[B] \Downarrow^{\mathsf{e}}_c$ and so $C[A] \Downarrow^{\mathsf{c}}_c \Leftrightarrow C[B] \Downarrow^{\mathsf{c}}_c$.

(3) Finally we show that the implication $\approx^{\mathsf{e}}_m \subsetneq \approx^{\mathsf{p}}_m \cap \approx^{\mathsf{c}}_m$ is strict, i.e., there exist processes A and B such that $A \approx^{\mathsf{c}}_m B$, $A \approx^{\mathsf{p}}_m B$ but $A \not\approx^{\mathsf{e}}_m B$. The processes defined in Fig. 7 witness this fact. They already were witness of the strict inclusion $\approx^{\mathsf{e}}_t \subsetneq \approx^{\mathsf{p}}_t \cap \approx^{\mathsf{c}}_t$ (see proof of Theorem 4) and since A and B are image finite, we know from Theorem 1 that may and trace equivalences between A and B coincide. □

4. Subclasses of processes for which (some of) the semantics coincide

As illustrated in previous sections, the presence of internal communications between public channels is the main issue when comparing the different semantics. Thus, the most natural class of processes on which the semantics coincide are processes where no internal communication on a public channel is possible.

Definition 7. Let P be a closed honest process. P is *internal communication free* if and only for all $P \overset{\mathsf{tr}}{\Rightarrow}_{\mathsf{c}} P' \xrightarrow{\tau}_{\mathsf{c}} P''$, the τ action in $P' \xrightarrow{\tau}_{\mathsf{c}} P''$ is not the application of the rule Comm.

We denote by $\mathcal{ICF}$ the set of internal communication free processes.

Lemma 1. *When restricted to* $\mathcal{ICF}$, $\approx^{s_1}_r = \approx^{s_2}_r$ *for* $r \in \{\ell, o, m, t\}$ *and* $s_1, s_2 \in \{\mathsf{c}, \mathsf{p}, \mathsf{e}\}$.

Proof. Immediate from the semantics. □

However, this class is very restrictive as it prevents any process of the form $\mathsf{out}^{\mathsf{ho}}(c, u).P \ \mid \mathsf{in}^{\mathsf{ho}}(c, x).Q(x)$. Therefore, we study in the rest of this section alternate classes of processes. The class of processes we study are mainly related to the notion of determinism: we first study the class of determinate processes, denoted $\mathcal{D}$, and then mainly restrict our attention to the case when the number of sessions is bounded. This is motivated by the fact that most tools able to verify these equivalences are restricted to a bounded number of sessions. We study three increasingly restrictive classes: (i) bounded determinate processes (denoted $\mathcal{BD}$), (ii) action-determinate processes (denoted $\mathcal{AD}$), and (iii) strong

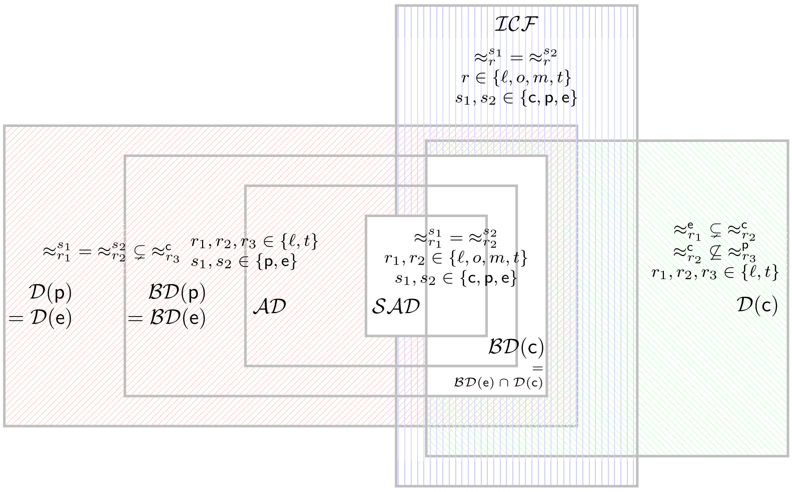

$\mathcal{ICF}$
$\approx_r^{s_1} = \approx_r^{s_2}$
$r \in \{\ell, o, m, t\}$
$s_1, s_2 \in \{\mathsf{c}, \mathsf{p}, \mathsf{e}\}$
$\approx_{r_1}^{s_1} = \approx_{r_2}^{s_2} \subsetneq \approx_{r_3}^{\mathsf{c}}$ $r_1, r_2, r_3 \in \{\ell, t\}$
$s_1, s_2 \in \{\mathsf{p}, \mathsf{e}\}$
$\mathcal{D}(\mathsf{p})$
$= \mathcal{D}(\mathsf{e})$
$\mathcal{BD}(\mathsf{p})$
$= \mathcal{BD}(\mathsf{e})$
$\mathcal{AD}$
$\approx_{r_1}^{s_1} = \approx_{r_2}^{s_2}$
$r_1, r_2 \in \{\ell, o, m, t\}$
$s_1, s_2 \in \{\mathsf{c}, \mathsf{p}, \mathsf{e}\}$
$\mathcal{SAD}$
$\mathcal{BD}(\mathsf{c})$
$=$
$\mathcal{BD}(\mathsf{e}) \cap \mathcal{D}(\mathsf{c})$
$\approx_{r_1}^{\mathsf{e}} \subsetneq \approx_{r_2}^{\mathsf{c}}$
$\approx_{r_2}^{\mathsf{c}} \not\subseteq \approx_{r_3}^{\mathsf{p}}$
$r_1, r_2, r_3 \in \{\ell, t\}$
$\mathcal{D}(\mathsf{c})$

Fig. 8. Summary of results for (bounded) determinate processes.

action determinate processes (denoted $\mathcal{SAD}$). As the definition of determinism depends on the semantics, we may add the semantics as a parameter, e.g., we write $\mathcal{D}(\mathsf{e})$ for the class of processes that are determinate in the eavesdrop semantics. Figure 8 provides an overview of the results of this section.

Finally, we also identify a new syntactic subclass of processes, called *I/O-unambiguous* and show relations among the equivalences for different semantics.

4.1. Determinate processes

4.1.1. Defining classes of determinate processes and their relations

In this section we define a subclass of determinate processes. These subclasses however depend on the semantics and therefore we also study the relations between these different subclasses. The notion of determinacy was defined in [15] for the classical semantics. It was shown that for determinate processes, observational and trace equivalence coincide. Intuitively, on determinate processes the attacker can determine, at each step of the execution, the position of the executed action in the process tree: this means that either the labels leading to the executed action differ, or, in case of two identical sequence of labels, the frames may be distinguished.

Definition 8 (Determinacy)**.** Let $s \in \{\mathsf{c}, \mathsf{p}, \mathsf{e}\}$. Let $\cong$ be an equivalence relation on closed honest extended processes. A closed honest extended process A is $\cong$-s-*determinate* if whenever $A \overset{\ell}{\Rightarrow}_s B$, $A \overset{\ell}{\Rightarrow}_s B'$ and $\phi(B) \sim \phi(B')$ then $B \cong B'$.

We denote by $\mathcal{D}(s, \cong)$ the set of closed honest extended processes that are $\cong$-s-determinate.

It was shown in [15] that $\mathcal{D}(\mathsf{c}, \approx_\ell^{\mathsf{c}}) = \mathcal{D}(\mathsf{c}, \approx_t^{\mathsf{c}})$. We can show that this equality also holds in the private and eavesdrop semantics.

Lemma 2. *For all $s \in \{\mathsf{c}, \mathsf{p}, \mathsf{e}\}$, $\mathcal{D}(s, \approx_\ell^s) = \mathcal{D}(s, \approx_t^s)$.*

Proof. The proof of [15, Lemma 2] literally holds for all semantics. □

$$P = !\mathsf{out}^{\mathsf{ho}}(c,a) \mid !\mathsf{in}^{\mathsf{ho}}(c,x).\mathsf{out}^{\mathsf{ho}}(d,a)$$
$$Q = !\mathsf{out}^{\mathsf{ho}}(c,a) \mid !\mathsf{in}^{\mathsf{ho}}(c,x) \mid !\mathsf{out}^{\mathsf{ho}}(d,a)$$

$$A = \mathsf{out}^{\mathsf{ho}}(c,a) \mid \mathsf{in}^{\mathsf{ho}}(c,x).\mathsf{out}^{\mathsf{ho}}(c,a) \qquad B = \nu s.\mathsf{out}^{\mathsf{ho}}(s,s) \mid \mathsf{in}^{\mathsf{ho}}(s,x).P \mid \mathsf{in}^{\mathsf{ho}}(s,x).Q$$

(a) $A \in \mathcal{D}(\mathsf{p})$ but $A \notin \mathcal{D}(\mathsf{c})$.

(b) $P, Q \in \mathcal{D}(\mathsf{c}) \cap \mathcal{D}(\mathsf{p})$, $P \approx_\ell^{\mathsf{c}} Q$ but $P \not\approx_t^{\mathsf{p}} Q$
$B \in \mathcal{D}(\mathsf{c})$ but $B \notin \mathcal{D}(\mathsf{p})$

Fig. 9. Examples differentiating classical and private semantics for determinate processes.

Thanks to the previous lemma, we may simply consider the set of s-determinate processes, denoted $\mathcal{D}(s)$, as the set of closed honest extended processes that are $\approx_\ell^s$-s-determinate or $\approx_t^s$-s-determinate coincide. It was also shown in [15] that when restricted to c-determinate processes, we have that $\approx_\ell^{\mathsf{c}}=\approx_t^{\mathsf{c}}$. Once again, this result directly extends to p-determinate and e-determinate processes.

Lemma 3. *When restricted to $\mathcal{D}(s)$, $\approx_\ell^s = \approx_t^s$ for $s \in \{\mathsf{c}, \mathsf{p}, \mathsf{e}\}$.*

Proof. The proof of [15, Theorem 2] literally holds for all semantics. □

The notion of determinacy depends on equivalences. Therefore one might expect the relations between determinate processes for different semantics to follow similar result as for the equivalences. However, we show that the sets of determinate processes coincide for eavesdrop and private semantics, while they are incomparable to the classic semantics

Lemma 4. $\mathcal{D}(\mathsf{p}) = \mathcal{D}(\mathsf{e})$, $\mathcal{D}(\mathsf{c}) \not\subseteq \mathcal{D}(\mathsf{p})$ *and* $\mathcal{D}(\mathsf{p}) \not\subseteq \mathcal{D}(\mathsf{c})$.

Proof sketch. We sketch the proof here. A more detailed version is available in Appendix F.

We start by showing that $\mathcal{D}(\mathsf{p}) \not\subseteq \mathcal{D}(\mathsf{c})$. Consider the process A displayed in Fig. 9a. $A \in \mathcal{D}(\mathsf{c})$ since $A \xrightarrow{\tau}_{\mathsf{c}} \mathsf{out}^{\mathsf{ho}}(c,a)$ by the rule COMM and $\mathsf{out}^{\mathsf{ho}}(c,a) \not\approx_\ell^{\mathsf{c}} A$. Moreover, $A \in \mathcal{D}(\mathsf{p})$ since for all tr, there is a unique A' such that $A \overset{\mathsf{tr}}{\Rightarrow}_{\mathsf{p}} A'$. Hence $\mathcal{D}(\mathsf{p}) \not\subseteq \mathcal{D}(\mathsf{c})$.

We now show that $\mathcal{D}(\mathsf{c}) \not\subseteq \mathcal{D}(\mathsf{p})$. Consider the process B displayed in Fig. 9b. Intuitively, the use of the private channel s in B encodes a non determinist choice between the two processes P and Q. We can show that $P, Q \in \mathcal{D}(\mathsf{c})$ which allows us to deduce that $B \in \mathcal{D}(\mathsf{c})$. However, $B \overset{\varepsilon}{\Rightarrow}_{\mathsf{p}} P$, $B \overset{\varepsilon}{\Rightarrow}_{\mathsf{p}} Q$ and $P \not\approx_t^{\mathsf{p}} Q$ imply $B \notin \mathcal{D}(\mathsf{p})$.

Let us show $\mathcal{D}(\mathsf{e}) \subseteq \mathcal{D}(\mathsf{p})$. Consider an honest closed process A such that $A \in \mathcal{D}(\mathsf{e})$. Let $A \overset{tr}{\Rightarrow}_{\mathsf{p}} A_1$ and $A \overset{tr}{\Rightarrow}_{\mathsf{p}} A_2$. By definition of the semantics, $A \overset{tr}{\Rightarrow}_{\mathsf{p}} A_i$ implies $A \overset{tr}{\Rightarrow}_{\mathsf{e}} A_i$, for $i = 1, 2$. Since $A \in \mathcal{D}(\mathsf{e})$, we deduce $A_1 \approx_\ell^{\mathsf{e}} A_2$. By applying Theorem 5, we obtain $A_1 \approx_\ell^{\mathsf{p}} A_2$ which concludes the proof of $\mathcal{D}(\mathsf{e}) \subseteq \mathcal{D}(\mathsf{p})$.

Finally, we need to show that $\mathcal{D}(\mathsf{p}) \subseteq \mathcal{D}(\mathsf{e})$. This part of the proof is detailed in Appendix F. □

4.1.2. Relations between semantics for determinate processes

We will now compare the equivalences when we restrict the processes to $\mathcal{D}(s)$ for a given semantics s. We start with the subclass $\mathcal{D}(\mathsf{p})$ (which coincides with $\mathcal{D}(\mathsf{e})$).

Theorem 7. *When restricted to $\mathcal{D}(\mathsf{p})$, we have $\approx_{r_1}^{s_1} = \approx_{r_2}^{s_2} \subsetneq \approx_{r_3}^{\mathsf{c}}$ for $s_1, s_2 \in \{\mathsf{p}, \mathsf{e}\}$, $r_1, r_2, r_3 \in \{\ell, t\}$.*

The proof is given in Appendix F.

Next, we consider the subclass $\mathcal{D}(\mathsf{c})$. We show that even for this subclass of processes, the equivalences for the classical semantics are not included in the other ones.

Lemma 5. *When restricted to* $\mathcal{D}(\mathsf{c})$*, we have* $\approx_r^{\mathsf{c}} \not\subseteq \approx_r^s$ *for* $s \in \{\mathsf{p}, \mathsf{e}\}$ *and* $r \in \{\ell, t\}$*.*

Proof. In the proof of Theorem 7 we showed that the processes P and Q displayed in Fig. 9 satisfy the following properties: $P, Q \in \mathcal{D}(\mathsf{c})$, $P \approx_\ell^{\mathsf{c}} Q$ and $P \not\approx_t^{\mathsf{p}} Q$. Note that we also have $P \not\approx_t^{\mathsf{e}} Q$. Hence P and Q allow us to prove that $\approx_r^{\mathsf{c}} \not\subseteq \approx_r^s$ for $s \in \{\mathsf{p}, \mathsf{e}\}$ and $r \in \{\ell, t\}$. □

4.2. *Determinacy for bounded processes*

As many verification tools [13,14,16,18,29] consider a bounded number of sessions we study in this section, notions of determinacy when restricted to processes without replication. We also consider the notion of *action-determinate* which was introduced in [10] as a subclass of determinate processes that enable partial order reductions that significantly speed-up verification. Finally, we discuss the notion of *strong* action-determinate processes: this class was introduced in several tools, as this property can easily be checked syntactically. Interestingly, for this class of action-determinate processes, all notions of equivalences and semantics coincide.

4.2.1. *Bounded determinate processes*

We investigate in this section whether additional relations hold between the semantics when restricted to bounded processes, i.e., processes without replication. In particular we show that when restricted to such bounded processes, a c-determinate P cannot have internal communication. However, we also show that even when restricted to bounded processes, $\approx_\ell^{\mathsf{p}}$ and $\approx_\ell^{\mathsf{c}}$ do not coincide.

We denote by $\mathcal{BD}(s)$ the set of bounded processes in $\mathcal{D}(s)$ for $s \in \{\mathsf{p}, \mathsf{c}, \mathsf{e}\}$.

Lemma 6. $\mathcal{BD}(\mathsf{c}) \subsetneq \mathcal{BD}(\mathsf{p}) = \mathcal{BD}(\mathsf{e})$ *and* $\mathcal{BD}(\mathsf{c}) \subsetneq \mathcal{ICF}$*.*

Proof. Note that Lemma 4 directly gives us $\mathcal{BD}(\mathsf{p}) = \mathcal{BD}(\mathsf{e})$. Moreover, consider the process A displayed in Fig. 9a. We already showed in Lemma 4 that $A \in \mathcal{D}(\mathsf{p})$ and $A \notin \mathcal{D}(\mathsf{c})$. Since A does not contain a replication, we deduce that $\mathcal{BD}(\mathsf{p}) \not\subseteq \mathcal{BD}(\mathsf{c})$.

Let us now show that $\mathcal{BD}(\mathsf{c}) \subseteq \mathcal{ICF}$. Let $A \in \mathcal{BD}(\mathsf{c})$. Assume by contradiction that $A \overset{\mathsf{tr}}{\Longrightarrow}_{\mathsf{c}} A_1 \xrightarrow{\tau}_{\mathsf{c}} A_2$ where the transition $A_1 \xrightarrow{\tau}_{\mathsf{c}} A_2$ is the application of the rule Comm. Hence $A_1 \equiv \nu\tilde{n}.(\mathsf{out}^{\mathsf{ho}}(c, u).P \mid \mathsf{in}^{\mathsf{ho}}(c, x).Q \mid R)$ and $A_2 = \nu\tilde{n}.(P \mid Q\{^u/_x\} \mid R)$ for some c, u, P, Q. Since $A \in \mathcal{BD}(\mathsf{c})$, we deduce that $A_1 \approx_t^{\mathsf{c}} A_2$. Consider the maximal trace tr_m of A_1 (the trace tr_m exists since A is bounded), i.e. $A_1 \overset{\mathsf{tr}_m}{\Longrightarrow}_{\mathsf{c}} A_1'$. Since $A_1 \approx_t^{\mathsf{c}} A_2$, the trace tr_m is also maximal for A_2 with $A_2 \overset{\mathsf{tr}_m}{\Longrightarrow}_{\mathsf{c}} A_2'$. But $A_1 \xRightarrow{\nu z.out(c,z).in(c,z)}_{\mathsf{c}} \nu\tilde{n}.(P \mid Q\{^u/_x\} \mid R \mid \{^u/_x\})$. Since $A_2 = \nu\tilde{n}.(P \mid Q\{^u/_x\} \mid R)$ and $A_2 \overset{\mathsf{tr}_m}{\Longrightarrow}_{\mathsf{c}} A_2'$, we deduce that $\nu\tilde{n}.(P \mid Q\{^u/_x\} \mid R \mid \{^u/_x\}) \overset{\mathsf{tr}_m}{\Longrightarrow}_{\mathsf{c}} A_2''$ for some A_2'' and so $\nu z.out(c, z).in(c, z).\mathsf{tr}_m$ is a trace of A_1 which contradicts the maximality of tr_m. Hence $\mathcal{BD}(\mathsf{c}) \subseteq \mathcal{ICF}$.

To see that the inclusion $\mathcal{BD}(\mathsf{c}) \subsetneq \mathcal{ICF}$ is strict, observe that for the process

$$P \hat{=} \mathsf{out}^{\mathsf{ho}}(c, a).\mathsf{out}^{\mathsf{ho}}(c, a_1) \mid \mathsf{out}^{\mathsf{ho}}(c, a).\mathsf{out}^{\mathsf{ho}}(c, a_2)$$

we have that $P \in \mathcal{ICF}$, but $P \notin \mathcal{BD}(\mathsf{c})$.

$$
\begin{aligned}
P &\mathrel{\hat{=}} \nu k_1, \ldots, k_7.(\mathsf{in}^{\mathsf{ho}}(c, x_1).R_1(x_1) \mid \mathsf{out}^{\mathsf{ho}}(c, k_1) \mid \mathsf{in}^{\mathsf{ho}}(d, x_2).\mathsf{if}\ x_2 = k_2\ \mathsf{then}\ \mathsf{out}^{\mathsf{ho}}(c, k_3))\\
Q &\mathrel{\hat{=}} \nu k_1, \ldots, k_7.(\mathsf{in}^{\mathsf{ho}}(c, x_1).R_1(x_1) \mid \mathsf{out}^{\mathsf{ho}}(c, k_1).\mathsf{in}^{\mathsf{ho}}(d, x_2).\mathsf{if}\ x_2 = k_2\ \mathsf{then}\ \mathsf{out}^{\mathsf{ho}}(c, k_3))
\end{aligned}
$$

where

$$
\begin{aligned}
R_1(x_1) &\mathrel{\hat{=}} \mathsf{if}\ x_1 = k_1\ \mathsf{then}\ \mathsf{out}^{\mathsf{ho}}(d, k_2).\mathsf{in}^{\mathsf{ho}}(c, x_3).\mathsf{if}\ x_3 = k_3\ \mathsf{then}\ R_3\ \mathsf{else}\ \mathsf{in}^{\mathsf{ho}}(d, x)\\
R_3 &\mathrel{\hat{=}} \mathsf{out}^{\mathsf{ho}}(c, k_4).\mathsf{in}^{\mathsf{ho}}(d, x_5).R_5(x_5) \mid \mathsf{in}^{\mathsf{ho}}(c, x_4).\mathsf{if}\ x_4 = k_4\ \mathsf{then}\ \mathsf{out}^{\mathsf{ho}}(d, k_5)\\
R_5(x_5) &\mathrel{\hat{=}} \mathsf{if}\ x_5 = k_5\ \mathsf{then}\\
&\quad\ \mathsf{in}^{\mathsf{ho}}(d, z).\\
&\quad\ (\mathsf{out}^{\mathsf{ho}}(c, k_6) \mid \mathsf{in}^{\mathsf{ho}}(c, x_6).\mathsf{if}\ x_6 = k_6\ \mathsf{then}\ \mathsf{out}^{\mathsf{ho}}(d, k_7).\mathsf{in}^{\mathsf{ho}}(c, x_3).\mathsf{in}^{\mathsf{ho}}(d, x))
\end{aligned}
$$

Fig. 10. $P \approx_\ell^{\mathsf{c}} Q$ but $P \not\approx_t^{\mathsf{p}} Q$.

Finally, let us prove $\mathcal{BD}(\mathsf{c}) \subseteq \mathcal{BD}(\mathsf{p})$. Let $A \in \mathcal{BD}(\mathsf{c})$, $A \overset{\mathsf{tr}}{\Rightarrow}_{\mathsf{p}} A_1$ and $A \overset{\mathsf{tr}}{\Rightarrow}_{\mathsf{p}} A_2$. Note that $A \overset{\mathsf{tr}}{\Rightarrow}_{\mathsf{p}} A_i$ implies $A \overset{\mathsf{tr}}{\Rightarrow}_{\mathsf{c}} A_i$, for $i = 1, 2$. As $A \in \mathcal{BD}(\mathsf{c})$, $A_1 \approx_\ell^{\mathsf{c}} A_2$. As $\mathcal{BD}(\mathsf{c}) \subseteq \mathcal{ICF}$, $A \in \mathcal{ICF}$ and so $A_1, A_2 \in \mathcal{ICF}$. By Lemma 1, we obtain $A_1 \approx_\ell^{\mathsf{p}} A_2$ which allows us to conclude. □

In particular, as $\mathcal{BD}(\mathsf{c}) \subset \mathcal{ICF}$ we directly have that all semantics coincide (Lemma 1), and by determinacy all equivalences coincide as well (Lemma 3).

Corollary 1. *When restricted to $\mathcal{BD}(\mathsf{c})$, we have that $\approx_{r_1}^{s_1} = \approx_{r_2}^{s_2}$ for $r_1, r_2 \in \{\ell, o, m, t\}$ and $s_1, s_2 \in \{\mathsf{c}, \mathsf{p}, \mathsf{e}\}$.*

We next investigate the relations when processes are restricted to the subclass $\mathcal{BD}(\mathsf{p})$ (which coincides with $\mathcal{BD}(\mathsf{e})$).

Theorem 8. *When restricted to $\mathcal{BD}(\mathsf{p})$, we have that $\approx_r^{\mathsf{p}} = \approx_r^{\mathsf{e}} \subsetneq \approx_r^{\mathsf{c}}$ for $r \in \{\ell, t\}$.*

The proof is given in Appendix G.

Action-determinate. As mentioned above, the class of action determinate processes is of interest for verification tools since it supports partial order reduction techniques [10] which speed-up verification by several orders of magnitude. Such techniques have been implemented in several verification tools such as APTE, AKISS and DeepSec.

Definition 9. Let P be a closed honest process. We say that P is action-determinate if $bn(P) \cap \mathcal{C}h = \emptyset$ and for all $P \overset{\mathsf{tr}}{\Rightarrow}_{\mathsf{c}} P'$, $P' \not\equiv \nu\tilde{k}.(\mathsf{out}^{\mathsf{ho}}(c, u_1).Q_1 \mid \mathsf{out}^{\mathsf{ho}}(c, u_2).Q_2|Q_3)$ and $P' \not\equiv \nu\tilde{k}.(\mathsf{in}^{\mathsf{ho}}(c, x_1).Q_1 \mid \mathsf{in}^{\mathsf{ho}}(c, x_2).Q_2|Q_3)$ for all $\tilde{k}, c, u_1, u_2, x_1, x_2, Q_1, Q_2, Q_3$.

We define $\mathcal{AD}$ to be the set of all action determinate processes.

Intuitively, a process is action determinate when there are never two similar available actions, i.e. two inputs or two outputs on the same channel. Note in particular that any (non-trivial) replicated process violates action-determinism.

We first show that the class of action determinate processes is strictly included in the class of determinate processes (for the private and, hence, eavesdropping semantics).

Lemma 7. $\mathcal{AD} \subsetneq \mathcal{BD}(\mathsf{p})$.

The proof is give in Appendix H.

We can now show that the relations among equivalences that did hold for the subclass $\mathcal{BD}(\mathsf{p})$ (Theorem 8) do also hold for the subclass $\mathcal{AD}$.

Theorem 9. *When restricted to $\mathcal{AD}$, we have that $\approx_r^{\mathsf{p}} = \approx_r^{\mathsf{e}} \subsetneq \approx_r^{\mathsf{c}}$ for $r \in \{\ell, t\}$.*

Note that, while the equality and inclusion is a direct corollary of Theorem 8 and Lemma 7, the fact that the inclusion is strict needs to be shown. The proof is given in Appendix I.

Strong action determinate. In the context of automated verification, deciding whether a process is action-determinate is still rather costly as it basically requires to verify a reachability property. We therefore introduce a stronger and more syntactical notion of action determinate, which is actually implemented in the verification tools AKISS and DeepSec. Intuitively, while action determinate processes never reach a situation where two "similar" actions are available, *strong* action-determinate processes verify that such similar actions never appear in parallel, syntactically.

We first define the set of action *skeletons* $\mathbb{S} = \{out(c), in(c) \mid c \in \mathcal{C}h\}$.

Definition 10 (strong action determinate)**.** The set $\mathcal{S}a(S)$ built on $S \subseteq \mathbb{S}$ is the smallest set of honest processes such that $\{^u/_x\}, 0 \in \mathcal{S}a(\emptyset)$ for all u, x and such that if $P \in \mathcal{S}a(S)$ and $Q \in \mathcal{S}a(S')$ then

- $\mathsf{out}^{\mathsf{ho}}(c, u).P \in \mathcal{S}a(\{out(c)\} \cup S)$ when $c \in \mathcal{C}h$
- $\mathsf{in}^{\mathsf{ho}}(c, x).P \in \mathcal{S}a(\{in(c)\} \cup S)$ when $c \in \mathcal{C}h$
- $\nu k;\ P \in \mathcal{S}a(S)$ when $\{in(k), out(k)\} \cap S = \emptyset$
- $\mathsf{if}\ u = v\ \mathsf{then}\ P\ \mathsf{else}\ Q \in \mathcal{S}a(S \cup S')$
- $\mathcal{S}a(P \mid Q) \in \mathcal{S}a(S \cup S')$ when $S \cap S' = \emptyset$

We define the set of strong action determinate process as $\mathcal{SAD} = \bigcup_{S \subseteq \mathbb{S}} \mathcal{S}a(S)$.

As the name indicates, it is easy to see that any strong action determinate process is also an action determinate process.

Lemma 8. $\mathcal{SAD} \subsetneq \mathcal{AD}$.

Proof. The implication follows directly from the definition, as any strongly action determinate process forbids two identic skeletons in parallel. To see that the implication is strict we observe that, for

$$P \mathrel{\hat{=}} \nu k.\ (\mathsf{out}^{\mathsf{ho}}(c, k) \mid \mathsf{in}^{\mathsf{ho}}(c, x).\mathsf{if}\ x = k\ \mathsf{then}\ \mathsf{out}^{\mathsf{ho}}(c, a))$$

we have that $P \in \mathcal{AD}$, but $P \notin \mathcal{SAD}$. □

While for action determinate processes we have that $\approx_\ell^{\mathsf{p}} \subsetneq \approx_\ell^{\mathsf{c}}$, we can show that for strong action determinate processes we actually have $\approx_\ell^{\mathsf{c}} \subseteq \approx_\ell^{\mathsf{p}}$.

Theorem 10. *When restricted to $\mathcal{SAD}$, we have $\approx_\ell^{\mathsf{c}} \subseteq \approx_\ell^{\mathsf{p}}$.*

Proof of Theorem 10 can be found in Appendix J.

This implies the following corollary stating that for strong action determinate processes, all semantics and equivalences coincide. This is particularly interesting as the AKISS and DeepSec tools check this condition. Moreover, it means that partial-order reduction optimizations, developed and shown correct for the private semantics [10], are correctly applied by these tools, regardless of the chosen semantics.

Corollary 2. *When restricted to $\mathcal{SAD}$, we have that $\approx_{r_1}^{s_1} = \approx_{r_2}^{s_2}$ for $r_1, r_2 \in \{\ell, o, m, t\}$ and $s_1, s_2 \in \{\mathsf{c}, \mathsf{p}, \mathsf{e}\}$.*

4.3. I/O-unambiguous processes

Restricting processes to action-determinate processes may sometimes be too restrictive. For instance, when verifying unlinkability and anonymity properties, two outputs by different parties should not be distinguishable due to the channel name. We therefore introduce another class of processes, that we call I/O-unambiguous for which we also show that the different semantics (although not the different equivalences) do coincide.

Intuitively, an io-unambiguous process forbids an output and input on the same public channel to follow each other directly (or possibly with only conditionals in between). For instance, we forbid processes of the form $\mathsf{out}^\theta(c,t).\mathsf{in}^\theta(c,x).P$, $\mathsf{out}^\theta(c,t).(\mathsf{in}^\theta(c,x).P \mid Q)$ as well as $\mathsf{out}^\theta(c,t).\mathsf{if}\ t_1 = t_2\ \mathsf{then}\ P\ \mathsf{else}\ \mathsf{in}^\theta(c,x).Q$. We however allow inputs and outputs on the same channel in parallel.

Definition 11. We define an honest extended process A to be I/O-unambiguous when $\mathsf{ioua}(A, \_) = \top$ where

$$
\begin{gathered}
\mathsf{ioua}(0,c) = \top \qquad \mathsf{ioua}(\{^u/_x\}, c) = \top \qquad \mathsf{ioua}(!P, c) = \mathsf{ioua}(P, c)\\
\mathsf{ioua}(A \mid B, c) = \mathsf{ioua}(A, c) \wedge \mathsf{ioua}(B, c) \qquad \mathsf{ioua}(\nu x.A, c) = \mathsf{ioua}(A, c)\\
\mathsf{ioua}(\nu n.A, c) = \begin{cases} \bot & \text{if } n \in \mathcal{C}h \\ \mathsf{ioua}(A, c) & \text{otherwise} \end{cases}\\
\mathsf{ioua}(\mathsf{if}\ u = v\ \mathsf{then}\ P\ \mathsf{else}\ Q, c) = \mathsf{ioua}(P, c) \wedge \mathsf{ioua}(Q, c)\\
\mathsf{ioua}(\mathsf{out}^\theta(d,u).P, c) = \begin{cases} \bot & \text{if } u \text{ is of channel type} \\ \mathsf{ioua}(P, d) & \text{otherwise} \end{cases}\\
\mathsf{ioua}(\mathsf{in}^\theta(d,x).P, c) = \begin{cases} \bot & \text{if } x \text{ is of channel type or } d = c \\ \mathsf{ioua}(P, \_) & \text{otherwise} \end{cases}
\end{gathered}
$$

Note that an I/O-unambiguous process does not contain private channels and always input/output base-type terms. We also note that a simple way to enforce that processes are I/O-unambiguous is to use disjoint channel names for inputs and outputs (at least in the same parallel thread).

Theorem 11. *When restricted to I/O-unambiguous processes, we have that $\approx_r^{\mathsf{p}} = \approx_r^{\mathsf{e}}$ but $\approx_r^{\mathsf{e}} \subsetneq \approx_r^{\mathsf{c}}$ for $r \in \{\ell, t\}$.*

Proof. From Theorems 5 and 4, we already know that $\approx_r^{\mathsf{e}} \subseteq \approx_r^{\mathsf{p}}$ and $\approx_r^{\mathsf{e}} \subseteq \approx_r^{\mathsf{c}}$. Hence, we only need to show that $\approx_r^{\mathsf{p}} \subseteq \approx_r^{\mathsf{e}}$ and $\approx_r^{\mathsf{p}} \subsetneq \approx_r^{\mathsf{c}}$. The latter is easily shown by noticing that the processes A and B in Fig. 6 are I/O-unambiguous. Thus, we focus on $\approx_r^{\mathsf{p}} \subseteq \approx_r^{\mathsf{e}}$.

We start by proving that for all I/O-unambiguous processes A, for all $A \overset{\mathsf{tr}}{\Rightarrow} A'$, we have that A' is I/O-unambiguous. Note that structural equivalence preserves I/O-unambiguity, *i.e.* for all extended processes A, B, for all channel name c, $A \equiv B$ implies $\mathsf{ioua}(A, c) = \mathsf{ioua}(B, c)$. Hence, we assume w.l.o.g. that a name is bound at most once and the set of bound and free names are disjoint.

Second, we show that for all I/O-unambiguous processes A, for all $A \xRightarrow{\nu z.out(c,z).in(c,z)}_{\mathsf{p}} A'$, we have that $\xRightarrow{\nu z.eav(c,z)}_{\mathsf{e}} A'$. To prove this property, denoted $\mathcal{P}$, let us assume w.l.o.g. that $A \xrightarrow{\nu z.out(c,z)}_{\mathsf{p}} A_1 \rightarrow^*_{\mathsf{p}} A_2 \xrightarrow{in(c,z)}_{\mathsf{p}} A'$. The transition $A \xrightarrow{\nu z.out(c,z)}_{\mathsf{p}} A_1$ indicates that $A \equiv \nu\tilde{n}.(\mathsf{out}^{\mathsf{ho}}(c, u).P \mid Q)$ and $A_1 \equiv \tilde{n}.(P \mid Q \mid \{^u/_z\})$ for some $P, Q, \tilde{n}, c, u$. Note that A is I/O-unambiguous, and hence $\mathsf{ioua}(P, c) = \top$.

As A is I/O-unambiguous implies that A does not contain private channels, we have that the rule applied in $A_1 \rightarrow^*_{\mathsf{p}} A_2$ is either the rule THEN or ELSE. Therefore, there exists P' and Q' such that $P \rightarrow^*_{\mathsf{p}} P'$, $Q \rightarrow^*_{\mathsf{p}} Q'$, $A_n \equiv \nu\tilde{n}.(P' \mid Q' \mid \{^u/_x\})$ and $\mathsf{ioua}(P', c) = \top$. Hence, we deduce that there exists Q_1, Q_2 such that $Q' \equiv \nu\tilde{m}.(\mathsf{in}^{\cdot}(c, x)Q_1 \mid Q_2)$ and $A' \equiv \nu\tilde{n}.\nu\tilde{m}.(P' \mid Q_1\{^u/_x\} \mid Q_2)$. We conclude the proof of this property by noticing that we can first apply on A the reduction rules of $Q \rightarrow^*_{\mathsf{p}} Q'$, then apply the rule C-EAV and finally apply the rules of $P \rightarrow^*_{\mathsf{p}} P'$.

(1) To prove $\approx^{\mathsf{p}}_t \subseteq \approx^{\mathsf{e}}_t$, we assume that A, B are two closed honest extended processes such that $A \approx^{\mathsf{p}}_t B$. For all $A \overset{\mathsf{tr}}{\Rightarrow}_{\mathsf{e}} A'$, it follows from the semantics that $A \overset{\mathsf{tr}_p}{\Rightarrow}_{\mathsf{p}} A'$ where tr_p is obtained by replacing in tr each $\nu z.eav(c, z)$ by $\nu z.out(c, z).in(c, z)$. Since $A \approx^{\mathsf{p}}_t B$, there exists B' such that $B \overset{\mathsf{tr}_p}{\Rightarrow}_{\mathsf{p}} B'$ and $\phi(A') \sim \phi(B')$. Thanks to the property $\mathcal{P}$, we conclude that $B \overset{\mathsf{tr}}{\Rightarrow}_{\mathsf{e}} B'$.

(2) To prove $\approx^{\mathsf{p}}_\ell \subseteq \approx^{\mathsf{e}}_\ell$, we assume that A, B are two closed honest extended processes such that $A \approx^{\mathsf{p}}_\ell B$ and let $\mathcal{R}$ be the relation witnessing this equivalence. We will show that $\mathcal{R}$ is also a labelled bisimulation in the eavesdropping semantics. Suppose $A\mathcal{R}B$.

- as $A \approx^{\mathsf{p}}_\ell B$, we have that $\phi(A) \sim \phi(B)$.
- if $A \xrightarrow{\tau}_{\mathsf{e}} A'$ then, as A is honest, $A \xrightarrow{\tau}_{\mathsf{p}} A'$. As $A \approx^{\mathsf{p}}_\ell B$ there exists B' such that $B \overset{\epsilon}{\Rightarrow}_{\mathsf{p}} B'$ and $A'\mathcal{R}B'$. As $\xrightarrow{\tau}_{\mathsf{p}} \subset \xrightarrow{\tau}_{\mathsf{e}}$, $B \overset{\epsilon}{\Rightarrow}_{\mathsf{e}} B'$
- if $A \xrightarrow{\ell}_{\mathsf{e}} A'$ then, as A is I/O-unambiguous, $A \overset{\mathsf{tr}}{\Rightarrow}_{\mathsf{e}} A'$ where $\mathsf{tr} = \nu z.out(c, z).in(c, z)$ when $\ell = \nu z.eav(c, z)$ else $\mathsf{tr} = \ell$. As $A \approx^{\mathsf{p}}_\ell B$, there exists B' such that $B \overset{\mathsf{tr}}{\Rightarrow}_{\mathsf{p}} B'$ and $A'\mathcal{R}B'$. When $\mathsf{tr} = \ell$, the definition of the semantics directly gives us $B \overset{\ell}{\Rightarrow}_{\mathsf{e}} B'$. When $\mathsf{tr} = \nu z.out(c, z).in(c, z)$, the property $\mathcal{P}$ gives us $B \overset{\ell}{\Rightarrow}_{\mathsf{e}} B'$. □

5. Different semantics in practice

As we have seen, in general, the three proposed semantics may yield different results. A conservative approach would consist in verifying always the eavesdropping semantics which is stronger than the two other ones, as shown before. However, this semantics seems also to be the least efficient one to verify. Moreover, partial-order reduction techniques that provide tremendous speed-ups were only developed for the private semantics.

We have implemented the three different semantics in the DeepSec tool. This allowed us to investigate the difference in results and performance between the semantics. In our experiments we considered several examples from DeepSec's example repository. We rely on the existing modelling and do not describe these protocols, as these details are not important for the observations we wish to make. All

Table 1

Experimental results

Property/Protocol	#roles	Single channel			I/O unambiguous		$\mathcal{SAD}$
		$\approx_t^{\mathsf{p}}$	$\approx_t^{\mathsf{e}}$	$\approx_t^{\mathsf{c}}$	$\approx_t^{\mathsf{p}} = \approx_t^{\mathsf{e}}$	$\approx_t^{\mathsf{c}}$	$\approx_t^{\mathsf{p}} = \approx_t^{\mathsf{e}} = \approx_t^{\mathsf{c}}$
Strong secrecy							
Denning–Sacco	6	33 s	2 m 7 s	1 m 58 s	9 s	35 s	<1 s
NSL	4	29 s	1 m	43 s	3 s	6 s	<1 s
Wide Mouth Frog	9	12 m 16 s	34 m 43 s	20 m 1 s	24 s	58 s	<1 s
Yahalom–Lowe	6	2 h 46 m	11 h 11 m	5 h 17 m	4 m 5 s	13 m 47 s	<1 s
Anonymity							
Passive Authentication	6	1 h 59 m	5 h 6 m	6 h 49 m	7 m 2 s	1 h 50 m	<1 s
Private Authentication	4	9 s	9 s	11 s	1 s	2 s	<1 s
Unlinkability							
Passive Authentication	6	3 h 15 m	7 h 15 m	11 h 6 m	10 m 30 s	2 h 49 m	<1 s
AKA	4	13 m	26 m 9 s	17 m 13 s	18 s	49 s	<1 s
Vote privacy							
Helios	10	10 m 9 s	19 m 10 s	14 m 50 s	–	–	–
Scytl	3	2 m 47 s	5 m 9 s	5 m 14 s	–	–	–

benchmarks, summarized in Table 1, were carried out on a machine with 20 Intel Xeon 3.10 GHz cores and 50 Gb of memory. The implementation and the specification files are available at [17].

The specifications we used for these experiments include verification of

- strong secrecy in several classical authentication protocols (Denning–Sacco, Needham–Schroeder–Lowe (NSL), Wide Mouth Frog, and Yahalom–Lowe protocols);
- anonymity of the Private Authentication protocol proposed by Abadi and Fournet [2];
- anonymity and unlinkability of the passive authentication protocol implemented in the European Passport protocol [5,23];
- unlinkability of the AKA protocol, deployed in 3G mobile telephony [7];
- vote privacy in the Helios e-voting protocol [4], and the e-voting protocol proposed by Scytl for elections in the Swiss Neuchâtel canton [19].

For all these examples we found that the results were unchanged, independent of the semantics. However, as expected, performance was generally better for the private semantics, and much better for strong action determinate processes, as this class allows for powerful partial-order reductions. The existing protocol encodings generally used a single public channel. To enforce membership in a particular subclass we had to use different channel names. Surprisingly, the use of distinct channels to enforce I/O-unambiguity, significantly enhances the tool's performance. We could not make the voting protocols I/O unambiguous and action determinate, because the encodings use private channels. In the absence of private channels using different channel names to enhance efficiency is tempting. One must however be careful as changing channel names changes the attacker's observation and may change the result. Typically, a single channel name models that the attacker does a priori not know which process sent a given message. Binding the channel name to the identity allows the attacker to know which host sent a message, but not necessarily which of the possibly multiple processes, e.g. sessions, on the given host. However, this modelling is typically not adequate when checking anonymity, or unlinkability, as it reveals the sender's identity. For such properties, it may be possible to use different channel names for each session, modelling that the adversary can distinguish different sessions, but not necessarily whether the same host executed one or several sessions. This is the encoding we used for verifying anonymity and unlinkability properties to enforce strong action determinism in the AKA and Passive Authentication protocols.

6. Conclusion

In this paper we investigated two families of Dolev–Yao models, depending on how the hypothesis that the *attacker controls the network* is reflected. While the two semantics coincide for reachability properties, they yield incomparable notions of behavioral equivalences, which have recently been extensively used to model privacy properties. The fact that forcing all communication to be routed through the attacker may diminish his distinguishing power may at first seem counter-intuitive. We also propose a third semantics, where internal communication among honest participants is permitted but leaks the message to the attacker. This new communication semantics entails strictly stronger equivalences than the two classical ones. We also identify several subclasses of protocols for which (some) semantics coincide. Finally, we implemented the three semantics in the DeepSec tool. Our experiments showed that the three semantics provide the same result on the case studies in the DeepSec example repository. However, the private semantics is slightly more efficient, as less interleavings have to be considered. Our results illustrate that behavioral equivalences are much more subtle than reachability properties and the need to carefully choose the precise attacker model.

Acknowledgments

We would like to thank Catherine Meadows and Stéphanie Delaune for interesting discussions, as well as the anonymous reviewers for their comments. This work has received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation program (grant agreement No 645865-SPOOC) and the ANR projects SEQUOIA ANR-14-CE28-0030-01 and TECAP ANR-17-CE39-0004-01.

Part of the work was carried out while the first author was a student at IIT Bombay during an internship at Inria Nancy Grand-Est.

Appendix A. Refining Theorem 3

We here give a more refined version of Theorem 3. In particular we show that the private and classical semantics are incomparable for trace equivalence and labelled bisimulation, even when restricted to processes that do not use else branches.

Theorem 12. *When restricted to processes without else branches, we have that* $\approx_r^{\mathsf{p}} \not\subseteq \approx_r^{\mathsf{c}}$ *and* $\approx_r^{\mathsf{c}} \not\subseteq \approx_r^{\mathsf{p}}$ *for* $r \in \{\ell, t\}$.

Proof. The fact that $\approx_r^{\mathsf{p}} \not\subseteq \approx_r^{\mathsf{c}}$ for $r \in \{\ell, t\}$ has already been shown in the proof of Theorem 3 as the processes A, B witnessing the result did not have else branches.

To show that $\approx_\ell^{\mathsf{c}} \not\subseteq \approx_\ell^{\mathsf{p}}$ we show that there exist processes A and B without else branches such that $A \approx_\ell^{\mathsf{c}} B$ and $A \not\approx_\ell^{\mathsf{p}} B$. Such processes are defined in Fig. 11. To see that $A \approx_\ell^{\mathsf{c}} B$ we first observe that

$$\begin{array}{l} A \mathrel{\hat{=}} \nu s.(\mathsf{in}^{\mathsf{ho}}(c, x).\mathsf{out}^{\mathsf{ho}}(c, s) \mid \mathsf{in}^{\mathsf{ho}}(c, y).P(y)) \\ B \mathrel{\hat{=}} \nu s.(\mathsf{in}^{\mathsf{ho}}(c, x).(\mathsf{out}^{\mathsf{ho}}(c, s) \mid \mathsf{in}^{\mathsf{ho}}(c, y).P(y))) \end{array}$$

where

$$P(y) \mathrel{\hat{=}} \mathsf{if}\ y = s\ \mathsf{then}\ \mathsf{in}^{\mathsf{ho}}(c, z).\mathsf{out}^{\mathsf{ho}}(c, s)$$

Fig. 11. A and B (without else branches) such that $A \approx^{\mathsf{c}}_{\ell} B$ and $A \not\approx^{\mathsf{p}}_{\ell} B$.

$$\begin{array}{l} A_i \mathrel{\hat{=}} \nu s_1.\nu s_2.(\,\mathsf{out}^{\mathsf{ho}}(c, h(s_1)) \mid \mathsf{out}^{\mathsf{ho}}(c, h(s_2)) \mid \\ \qquad\qquad \mathsf{in}^{\mathsf{ho}}(d, x).(\mathsf{if}\ x = h(s_1)\ \mathsf{then}\ Q_i \mid \mathsf{if}\ x = h(s_2)\ \mathsf{then}\ P_2) \end{array}$$

where $Q_1 \hat{=} P_1$, $Q_2 \hat{=} P_2$ and

$$\begin{array}{l} P_1 \mathrel{\hat{=}} \mathsf{out}^{\mathsf{ho}}(e, a) \\ P_2 \mathrel{\hat{=}} \mathsf{out}^{\mathsf{ho}}(f, a).\mathsf{out}^{\mathsf{ho}}(e, a) \mid \mathsf{in}^{\mathsf{ho}}(f, x) \end{array}$$

Fig. 12. A_1 and A_2 such that $A_1 \approx^{\mathsf{c}}_{t} A_2$, but $A_1 \not\approx^{\mathsf{p}}_{t} A_2$.

the only first possible action from A or B is an input. In particular, given a term t, there is a unique B' such that $B \xrightarrow{in(c,t)} B'$ where $B' = \nu s.(\mathsf{out}^{\mathsf{ho}}(c, s) \mid \mathsf{in}^{\mathsf{ho}}(c, y).P(y))$. On the other hand, if $A \xrightarrow{in(c,M)} A'$ then either $A' = B'$ or $A' = A''$ where $A'' \mathrel{\hat{=}} \nu s.(\mathsf{in}^{\mathsf{ho}}(c, x).\mathsf{out}^{\mathsf{ho}}(c, s) \mid P(t))$. Therefore, to complete the proof, we only need to find B'' such that $B \overset{in(c,t)}{\Longrightarrow} B''$ and $A'' \approx^{\mathsf{c}}_{\ell} B''$. Such process can be obtain by applying an internal communication on B', *i.e.* $B \xrightarrow{in(c,t)}_{\mathsf{c}} B' \xrightarrow{\tau} \nu s.P(s)$. Note that $t \neq s$ since s is bound, meaning that $P(t) \approx^{\mathsf{c}}_{\ell} 0$. Moreover, $P(s) \approx^{\mathsf{c}}_{\ell} \mathsf{in}^{\mathsf{ho}}(c, x).\mathsf{out}^{\mathsf{ho}}(c, s)$. This allows us to conclude that $\nu s.P(s) \approx^{\mathsf{c}}_{\ell} A''$.

To see that $A \not\approx^{\mathsf{p}}_{\ell} B$ we first observe that when $A \xrightarrow{in(c,t)}_{\mathsf{p}} A''$, B can only mimic A by preforming the transition $B \xrightarrow{in(c,t)} B'$. We conclude as $B' \xrightarrow{\nu z.out(c,z)}_{\mathsf{p}} \nu s.(\mathsf{in}^{\mathsf{ho}}(c, y).P(y) \mid \{{}^{s}/_{z}\})$ and $A'' \not\xrightarrow{\nu z.out(c,z)}_{\mathsf{p}}$.

We next show that there also exist A_1 and A_2 such that $A_1 \approx^{\mathsf{c}}_{t} A_2$, but $A_1 \not\approx^{\mathsf{p}}_{t} A_2$.

We define such processes in Fig. 12. Using the DeepSec tool we have shown that indeed $A_1 \approx^{\mathsf{c}}_{t} A_2$ and $A_1 \not\approx^{\mathsf{p}}_{t} A_2$. The main argument why the result holds is that P_1 is trace included in P_2 in the classical semantics (as the output on channel f can be made silent through an internal communication) while this is not the case in the private semantics. □

Appendix B. Proof of Proposition 2

Definition 12. We say that a plain process P (resp. extended process P) is name-cleaned if P is of the form $P_1 \mid \ldots \mid P_m$ and every P_i is not of the form $\nu k.B'$ with k a name or variable of any type.

Lemma 9. *Let A be an extended process. There exist a sequence of names and variables $\tilde{k}$ and a name-cleaned extended process A' such that $A \equiv \nu\tilde{k}.A'$.*

Proof. Direct from the definition of structural equivalence. □

Proposition 2. *For all closed honest plain processes A, for all $s \in \{\mathsf{c}, \mathsf{e}, \mathsf{p}\}$, $A \downdownarrows_c^s$ iff there exists an attacker plain process I^s such that $I^s \mid A \Downarrow_c^{s,\mathsf{ho}}$.*

Proof. We will prove that $A \downdownarrows_c^s$ implies there exists an attacker plain process I^s such that $C^s[A] \Downarrow_c^s$ for $s \in \{\mathsf{c}, \mathsf{e}, \mathsf{p}\}$ by constructing I^s.

Let us first focus on $s = \mathsf{c}$. Since $A \downdownarrows_c^{\mathsf{c}}$, we know that there exist A', t and $\mathsf{tr} \in (\mathcal{A} \setminus \{\tau\})^*$ such that $A \overset{\mathsf{tr}}{\Rightarrow}_{\mathsf{c}} A'$, $c \notin bn(\mathsf{tr})$ and $out(c, t) \in \mathsf{tr}$. Note that we can assume w.l.o.g. that no name in tr is bound twice and bound names in tr are distinct from free names that occurs in A and tr.

Let $\{a_1, \ldots, a_k\}$ be all the channel names that occur in tr (bound or free). To each $a_1, \ldots, a_k$, we associate a variable of channel type $x_{a_1}, \ldots, x_{a_k}$. Given a subset $S \subseteq \{a_1, \ldots, a_k\}$, we denote by $\sigma(S)$ the substitution $\{x_a \to a \mid a \in S\}$. We define I^{c} such that $I^{\mathsf{c}} = \mathsf{Q_c}(\mathsf{tr}, \sigma(fc(\mathsf{tr})))$ where $\mathsf{Q_c}(\mathsf{tr}, \sigma)$ is defined by induction on tr as follows:

- if $\mathsf{tr} = \epsilon$ then $\mathsf{Q_c}(\mathsf{tr}, \sigma) = 0$;
- if $\mathsf{tr} = in(a, M).\mathsf{tr}'$ then $\mathsf{Q_c}(\mathsf{tr}, \sigma) = \mathsf{out}^{\mathsf{at}}(x_a\sigma, M).\mathsf{Q_c}(\mathsf{tr}', \sigma)$;
- if $\mathsf{tr} = out(a, c).\mathsf{tr}'$ with c of channel-type then
$$\mathsf{Q_c}(\mathsf{tr}, \sigma) = \mathsf{in}^{\mathsf{at}}(x_a\sigma, y).\mathsf{Q_c}(\mathsf{tr}', \sigma)$$
where y is fresh variable of channel type;
- if $\mathsf{tr} = \nu x.out(a, x).\mathsf{tr}'$ and x is of base type then
$$\mathsf{Q_c}(\mathsf{tr}, \sigma) = \mathsf{in}^{\mathsf{at}}(x_a\sigma, x).\mathsf{Q_c}(\mathsf{tr}', \sigma)$$
- if $\mathsf{tr} = \nu c.out(a, c).\mathsf{tr}'$ and c is of channel type then
$$\mathsf{Q_c}(\mathsf{tr}, \sigma) = \mathsf{in}^{\mathsf{at}}(x_a\sigma, x_c).\mathsf{Q_c}(\mathsf{tr}', \sigma)$$

Since $A \overset{\mathsf{tr}}{\Rightarrow}_{\mathsf{c}} A'$, there exist $A_0, \ldots, A_n$ and $\ell_1, \ldots, \ell_N$ such that $A' = A_N$, $A = A_0$ and $A_0 \xrightarrow{\ell_1}_{\mathsf{c}} A_1 \xrightarrow{\ell_2}_{\mathsf{c}} \ldots \xrightarrow{\ell_N}_{\mathsf{c}} A_N$. We can show by induction that for all $n \leqslant N$, there exist a plain process Q_n and two sequences of names $\tilde{y}_n, \tilde{r}_n$ such that:

- $I^{\mathsf{c}}A \to_{\mathsf{c}}^* \nu\tilde{y}_n.\nu\tilde{r}_n.(A_n \mid Q_n)$
- $\tilde{r}_n$ is the sequence of bounded channel names in $\ell_1 \cdot \cdots \cdot \ldots \ell_{n-1}$
- $\tilde{y}_n \subseteq \mathrm{dom}(\phi(A_n))$
- tr_n is the sequence $\ell_n \cdot \ldots \cdot \ell_N$ where the τ action are removed
- $Q_n = \mathsf{Q_c}(\mathsf{tr}_n, \sigma(fc(\mathsf{tr}_n)))$

To conclude this proof, recall that $out(c, t) \in \mathsf{tr}$ and $c \notin bn(\mathsf{tr})$ so there exists $n \leqslant N$ such that $\ell_n = out(c, t)$ or $\ell_n = \nu t.out(c, t)$. But since $A_{n-1} \xrightarrow{\ell_n}_{\mathsf{c}} A_n$ and $A_{n-1} \equiv \nu\tilde{k}_{n-1}.B_{n-1}$ with B_{n-1} being name-cleaned, we deduce that there exist P, R such that $B_{n-1} = \mathsf{out}^{\mathsf{ho}}(c, t).P \mid R$ and $c \notin \tilde{k}_{n-1}$.

Therefore, $I^{\mathsf{c}} \mid A \rightarrow_{\mathsf{c}}^{*} \nu\tilde{y}_{n-1}.\nu\tilde{r}_{n-1}.\nu\tilde{k}_{n-1}.(\mathsf{out}^{\mathsf{ho}}(c,t).P \mid R \mid Q_{n-1})$. Note that $\tilde{y}_{n-1} \subseteq \mathrm{dom}(\phi(B_{n-1}))$ hence $c \notin \tilde{y}_{n-1}$. Moreover, we assumed that $c \notin bn(\mathsf{tr})$ hence $c \notin \tilde{r}_{n-1}$ by definition of $\tilde{r}_{n-1}$. It allows us to conclude $I^{\mathsf{c}} \mid A \Downarrow_{c}^{\mathsf{c},\mathsf{ho}}$.

The proof for the other two semantics is very similar. First, the construction of the context changes to adapt the changes in the labeled semantics. Second, we prove a slightly different property on the traces to account the presence of opened channels that are generated by the rule C-OPEN. The rest stay the same (up to renaming of c into p and e respectively).

Concerning the semantics private, we define $I^{\mathsf{p}} \hat{=} I^{\mathsf{c}}$ and we can prove the following property: For all $n \leqslant N$, there exist two extended processes Q_n, R_n and thwo sequences of names $\tilde{y}_n$, $\tilde{r}_n$ such that:

- $C_{\mathsf{p}}[A] \rightarrow_{\mathsf{p}}^{*} \nu\tilde{y}_n.\nu\tilde{r}_n.(A_n \mid Q_n \mid R_n)$
- $\tilde{r}_n$ is the sequence of bounded channel names in $\ell_1 \cdot \cdots . . . \ell_{n-1}$
- $R_n \hat{=} \omega c_1 \mid \ldots \mid \omega c_m$ for some $c_1, \ldots, c_m$ such that $\tilde{r}_n \subseteq \{c_1, \ldots, c_m\}$
- $\tilde{y}_n \subseteq \mathrm{dom}(\phi(B_n))$
- tr_n is the sequence $\ell_n \cdot \ldots \cdot \ell_N$ where the τ action are removed
- $Q_n = \mathsf{Q}_{\mathsf{c}}(\mathsf{tr}_n, \sigma(fc(\mathsf{tr}_n)))$

Notice that the presence of R_n is the only difference between the property in the classical and private semantics. This is the consequence of the application of the rule C-OPEN that introduces opened channels ωc_1 and that we apply when the trace contains labeled transitions $out(c,d)$ or $\nu d.out(c,d)$.

For the eavesdropping semantics, we can prove the same property as for private semantics (up to renaming of p into e) but we need to modify the context as follows. We define $C_{\mathsf{e}}[\_]$ such that $I^{\mathsf{e}}[\_] \hat{=} \mathsf{Q}_{\mathsf{e}}(\mathsf{tr}, \sigma)$ is defined by induction on tr as follows:

- if $\mathsf{tr} = \epsilon$ then $\mathsf{Q}_{\mathsf{e}}(\mathsf{tr}, \sigma) = 0$;
- if $\mathsf{tr} = in(a, M).\mathsf{tr}'$ then $\mathsf{Q}_{\mathsf{e}}(\mathsf{tr}, \sigma) = \mathsf{out}^{\mathsf{at}}(x_a\sigma, M).\mathsf{Q}_{\mathsf{e}}(\mathsf{tr}', \sigma)$;
- if $\mathsf{tr} = out(a, c).\mathsf{tr}'$ with c of channel-type then

  $$\mathsf{Q}_{\mathsf{e}}(\mathsf{tr}, \sigma) = \mathsf{in}^{\mathsf{at}}(x_a\sigma, y).\mathsf{Q}_{\mathsf{e}}(\mathsf{tr}', \sigma)$$

  where y is fresh variable of channel type;
- if $\mathsf{tr} = \nu x.out(a, x).\mathsf{tr}'$ and x is of base type then

  $$\mathsf{Q}_{\mathsf{e}}(\mathsf{tr}, \sigma) = \mathsf{in}^{\mathsf{at}}(x_a\sigma, x).\mathsf{Q}_{\mathsf{e}}(\mathsf{tr}', \sigma)$$

- if $\mathsf{tr} = \nu c.out(a, c).\mathsf{tr}'$ and c is of channel type then

  $$\mathsf{Q}_{\mathsf{e}}(\mathsf{tr}, \sigma) = \mathsf{in}^{\mathsf{at}}(x_a\sigma, x_c).\mathsf{Q}_{\mathsf{e}}(\mathsf{tr}', \sigma)$$

- if $\mathsf{tr} = eav(a, c).\mathsf{tr}'$ with c of channel-type then

  $$\mathsf{Q}_{\mathsf{e}}(\mathsf{tr}, \sigma) = \mathsf{eav}(x_a\sigma, y).\mathsf{Q}_{\mathsf{e}}(\mathsf{tr}', \sigma)$$

  where y is fresh variable of channel type;
- if $\mathsf{tr} = \nu x.eav(a, x).\mathsf{tr}'$ and x is of base type then

  $$\mathsf{Q}_{\mathsf{e}}(\mathsf{tr}, \sigma) = \mathsf{eav}(x_a\sigma, x).\mathsf{Q}_{\mathsf{e}}(\mathsf{tr}', \sigma)$$

- if $\mathsf{tr} = \nu c.eav(a, c).\mathsf{tr}'$ and c is of channel type then

$$\mathsf{Q_e}(\mathsf{tr}, \sigma) = \mathsf{eav}(x_a\sigma, x_c).\mathsf{Q_e}(\mathsf{tr}', \sigma)$$

Let us now focus on the other implications, that are: if there exists an attacker plain process I^s such that $I^s \mid A \Downarrow_c^{s,\mathsf{ho}}$ then $A \downdownarrows_c^s$ for $s \in \{\mathsf{c}, \mathsf{e}, \mathsf{p}\}$. By Lemma 9, we can assume w.l.o.g. that $I^s = \nu\tilde{k}.D$ for some name-cleaned plain process D and some sequence of names and variables of any type $\tilde{k}$. We now prove that for all $I^s \mid A \rightarrow_s^* B$, there exist an attacker evaluation context $C'[\_] = \nu\tilde{k}'.(D' \mid \_)$ with D' name-cleaned, an honest extended process A' and tr such that $C'[\_]$ is s-closing for A, $B \equiv C'[A']$ and $A \overset{\mathsf{tr}}{\Rightarrow}_s A'$. We first focus on $s = \mathsf{c}$. Note that it is not necessary to prove the property name-cleaned since it is implied by Lemma 9.

We prove this result by induction on the number of reduction rules in $I^{\mathsf{c}} \mid A \rightarrow_{\mathsf{c}}^* B$.

Base case: By structural equivalence, there exists $\tilde{k}'$ and D' such that $I \mid A \equiv \nu\tilde{k}'.(D' \mid A)$. Moreover, since $fv(A) = \emptyset$, $\tilde{k}'.(D' \mid \_)$ is closing for A and so the base case holds.

Inductive step $C^{\mathsf{c}}[A] \rightarrow_{\mathsf{c}}^* B' \rightarrow_{\mathsf{c}} B$: By our inductive hypothesis, we know that there exist $C'[\_] = \nu\tilde{k}'.(D' \mid \_)$, and honest extended process A' and tr such that $C'[\_]$ is c-closing for A', $B' \equiv C'[A']$ and $A \overset{\mathsf{tr}}{\Rightarrow}_{\mathsf{c}} A'$. Note that due to the structural equivalence, we can assume w.l.o.g. that $A' = \nu\tilde{r}.P$ where P is name-cleaned. Moreover, since $B' \rightarrow_{\mathsf{c}} B$ and $B' \equiv C'[A']$, we deduce that $C'[A'] \rightarrow_{\mathsf{c}} B$. Let us do a case analysis on the rule applied.

Case 1, internal reduction on A', i.e. there exists A'' such that $A' \xrightarrow{\tau}_{\mathsf{c}} A''$ and $B \equiv C'[A'']$. In such a case, we have that $C'[A'] \xrightarrow{\tau}_{\mathsf{c}} C'[A'']$. Moreover, since $A \overset{\mathsf{tr}}{\Rightarrow}_{\mathsf{c}} A'$ then we directly obtain that $A \overset{\mathsf{tr}}{\Rightarrow}_{\mathsf{c}} A''$ and so the result holds.

Case 2, internal reduction on C', i.e. there exists D'' such that $D' \xrightarrow{\tau}_{\mathsf{c}} D''$ and $B \equiv \nu\tilde{k}'.(D'' \mid A')$. By the structural equivalence, we know that there exist $\tilde{k}''$ and D''' such that D'' is named-cleaned and $\nu\tilde{k}'.(D'' \mid A') \equiv \nu\tilde{k}''.(D''' \mid A')$. Therefore, we can define $C''[\_] = \nu\tilde{k}''.(D''' \mid \_)$ and obtain that $C'[A'] \xrightarrow{\tau}_{\mathsf{c}} C''[A']$. Since $A \overset{\mathsf{tr}}{\Rightarrow}_{\mathsf{c}} A'$, the result holds.

Case 3, rule COMM *between C' (input) and A' (output), i.e. $D' = \mathsf{in}^{\mathsf{at}}(c, x).D_1 \mid D_2$, $A' = \nu\tilde{r}.(\mathsf{out}^{\mathsf{ho}}(c, u).P_1 \mid P_2)$ and $B \equiv \nu\tilde{k}'.\nu\tilde{r}.(D_1\{^u/_x\} \mid D_2 \mid P_1 \mid P_2)$ (We assume w.l.o.g. that the names and variables in $\tilde{r}$ are not in D').* Note that in such a case, $c \notin \tilde{r}$. We do a case analysis on u.

- Case 3.a, $u \in \mathcal{C}h \cap \tilde{r}$: Let us redenote $\nu\tilde{r}$ as $\nu\tilde{r}'.\nu u$. Thus $A' \xrightarrow{\nu u.out(c,u)}_{\mathsf{c}} \nu\tilde{r}'.(P_1 \mid P_2)$. Hence, since the names and variables in $\tilde{r}$ are not in D', we obtain that $B \equiv \nu\tilde{k}'.\nu u.(D_1\{^u/_x\} \mid D_2 \mid \nu\tilde{r}'.(P_1 \mid P_2))$. Hence, by denoting $C''[\_] = \nu\tilde{k}'.\nu u.(D_1\{^u/_x\} \mid D_2 \mid \_)$ and $A'' = \nu\tilde{r}'.(P_1 \mid P_2)$, the result hold.
- Case 3.b, $u \in \mathcal{C}h$ but $u \notin \tilde{r}$. In such a case, $A' \xrightarrow{out(c,u)}_{\mathsf{c}} \nu\tilde{r}.(P_1 \mid P_2)$. Hence, since the names and variables in $\tilde{r}$ are not in D', we obtain that $B \equiv \nu\tilde{k}'.(D_1\{^u/_x\} \mid D_2 \mid \nu\tilde{r}.(P_1 \mid P_2))$. By denoting $C''[\_] = \nu\tilde{k}'.(D_1\{^u/_x\} \mid D_2 \mid \_)$ and $A'' = \nu\tilde{r}.(P_1 \mid P_2)$, the result holds.
- Case 3.c, $u \notin \mathcal{C}h$: In such a case, $A' \xrightarrow{\nu y.out(c,y)}_{\mathsf{c}} \nu\tilde{r}.(P_1 \mid P_2 \mid \{^u/_y\})$ with $y \notin fv(A') \cup v(u)$. Note we can take y such that $y \notin fv(C'[A']) \cup bn(C'[A'])$. Note that $B \equiv \nu\tilde{k}'.\nu\tilde{r}.(D_1\{^u/_x\} \mid D_2 \mid P_1 \mid P_2)$. By definition of the structural equivalence and since we took $y \notin fv(C'[A']) \cup bn(C'[A'])$, we deduce that $B \equiv \nu\tilde{k}'.\nu y.\nu\tilde{r}.(D_1\{^y/_x\} \mid D_2 \mid P_1 \mid P_2 \mid \{^u/_y\})$. Lastly, since the names and variables in $\tilde{r}$ are not in D', we deduce that $B \equiv \nu\tilde{k}'.\nu y.(D_1\{^y/_x\} \mid D_2 \mid \nu\tilde{r}.(P_1 \mid P_2 \mid \{^u/_y\}))$. By denoting $C''[\_] = \nu\tilde{k}'.\nu y.(D_1\{^y/_x\} \mid D_2 \mid \_)$ and $A'' = \nu\tilde{r}.(P_1 \mid P_2 \mid \{^u/_y\})$, the result holds.

Case 4, rule COMM *between A' (input) and C' (output), i.e. $D' = \mathsf{out}^{\mathsf{at}}(c, u).D_1 \mid D_2$, $A' = \nu\tilde{r}.(\mathsf{in}^{\mathsf{ho}}(c, x).P_1 \mid P_2)$ and $B \equiv \nu\tilde{k}'.\nu\tilde{r}.(D_1 \mid D_2 \mid P_1\{^{u}/_{x}\} \mid P_2)$ (We assume w.l.o.g. that the names and variables in $\tilde{r}$ are not in D').* Note that in such a case, $c \notin \tilde{r}$. Moreover, we know that the names and variables in $\tilde{r}$ are not in D', meaning that the names and variables in $\tilde{r}$ does not occur in u. Hence, $A' \xRightarrow{in(c,u)}_{\mathsf{c}} \nu\tilde{r}.(P_1\{^{u}/_{x}\} \mid P_2)$. Once again due to the fact the names and variables in $\tilde{r}$ are not in D', we obtain that $B'' \equiv \nu\tilde{k}'.(D_1 \mid D_2 \mid \nu\tilde{r}.(P_1\{^{u}/_{x}\} \mid P_2))$. By denoting $C''[\_] = \nu\tilde{k}'.(D_1 \mid D_2 \mid \_)$ and $A'' = \nu\tilde{r}.(P_1\{^{u}/_{x}\} \mid P_2)$, the result holds.

We have concluded the proof of the property: for all $C^{\mathsf{c}}[A] \rightarrow_{\mathsf{c}}^{*} B$, there exist an evaluation attacker context $C'[\_] = \nu\tilde{k}'.(D' \mid \_)$ with D' name-cleaned, an honest extended process A' and tr such that $C'[\_]$ is c-closing for A', $B \equiv C'[A']$ and $A \overset{\mathsf{tr}}{\Rightarrow}_{\mathsf{c}} A'$. It remains to prove this result for $s = \mathsf{e}$ and $s = \mathsf{p}$. Let us focus first on the case $s = \mathsf{p}$. The proof is in fact similar to the case $s = \mathsf{c}$. Notice that the case of the rule C-ENV correspond to either Case 2, 3.c or 4 when u is of base type. Hence it remains the case of the rules C-PRIV and C-OPEN.

Case 5, rule C-OPEN *between C' (input) and A' (output), i.e. $D' = \mathsf{in}^{\theta}(c, x).D_1 \mid D_2$, $A' = \nu\tilde{r}.(\mathsf{out}^{\mathsf{ho}}(c, d).P_1 \mid P_2)$ and $B \equiv \nu\tilde{k}'.\nu\tilde{r}.(D_1\{^{d}/_{x}\} \mid D_2 \mid P_1 \mid P_2 \mid \omega d)$.* Lets us do a case analysis on whether (5.a) $d \in \tilde{r}$ or (5.b) $d \notin \tilde{r}$. Note that Case (5.a) is in fact almost identical to Case (3.a) and that the result holds with $C''[\_] = \nu\tilde{k}'.\nu d.(D_1\{^{u}/_{x}\} \mid D_2 \mid \omega d \mid \_)$ and $A'' = \nu\tilde{r}'.(P_1 \mid P_2)$ with $\nu\tilde{r} = \nu\tilde{r}'.\nu d$. Furthermore, note that Case (5.b) is also very similar to Case (3.b) and that the result holds with $C''[\_] = \nu\tilde{k}'.(D_1\{^{d}/_{x}\} \mid D_2 \mid \omega d\_)$ and $A'' = \nu\tilde{r}.(P_1 \mid P_2)$. Notice that in both case $C''[\_]$ is indeed p-closing for A''.

Case 6, rule C-OPEN *between A' (input) and C' (output), i.e. $D' = \mathsf{out}^{\theta}(c, d).D_1 \mid D_2$, $A' = \nu\tilde{r}.(\mathsf{in}^{\mathsf{ho}}(c, x).P_1 \mid P_2)$ and $B \equiv \nu\tilde{k}'.\nu\tilde{r}.(D_1 \mid D_2 \mid P_1\{^{d}/_{x}\} \mid P_2 \mid \omega d)$.* This case if very similar to Case 4 when u is of channel type and the result holds with $C''[\_] = \nu\tilde{k}'.(D_1 \mid D_2 \mid \omega d \mid \_)$ and $A'' = \nu\tilde{r}.(P_1\{^{d}/_{x}\} \mid P_2)$.

Case 7, rule C-PRIV *with a communication on a channel c.* Notice that this rule is in fact partially covered by the beginning of the proof. Indeed, Case 1 and 2 cover the cases where c is not in $\tilde{k}'$. Therefore, we only need to focus on the case where the private channel is in $\tilde{k}'$, *i.e.* $\nu\tilde{k}' = \nu\tilde{k}''.\nu c$ for some $\tilde{k}''$. We know that $C'[\_]$ is p-closing for A'. Hence since c is a channel bound in $C'[\_]$ whose scope includes $\_$, we deduce that if $c \in fn(A)$ then ωc is also in the scope of c. But according to the definition of the rule, we know that ωc is not in the scope of νc. Moreover, if the output or input is done by A' then it would implies that $c \in fn(A)$. Thus, this allows us to deduce that this both output and input are tagged with at, meaning that there exists D'' such that $\nu c.(D' \mid A') \xrightarrow{\tau}_{\mathsf{p}} \nu c.(D'' \mid A')$ and $B \equiv \nu\tilde{k}''.\nu c.(D'' \mid A')$. In such a case, by denoting $C''[\_] = \nu\tilde{k}''.\nu c.(D'' \mid \_)$ and $A'' = A'$, the result directly holds.

We have concluded the proof of the property for $s = \mathsf{p}$ hence it remains the case $s = \mathsf{e}$. Once, again several cases are already covered since $\xrightarrow{\ell}_{\mathsf{p}} \subset \xrightarrow{\ell}_{\mathsf{e}}$. Hence we only need to focus on the cases of the rules C-EAV and C-OEAV:

Case 8, rule C-EAV, i.e. $A' = \nu\tilde{r}.(\mathsf{out}^{\mathsf{ho}}(c, u).P_1 \mid \mathsf{in}^{\mathsf{ho}}(c, x).P_2 \mid P_3)$*, $D'' = \mathsf{eav}(c, y).Q_1 \mid Q_2$), $B \equiv \nu\tilde{k}'.\nu\tilde{r}.(Q_1\{^{u}/_{y}\} \mid Q_2 \mid P_1 \mid P_2\{^{u}/_{x}\} \mid P_3)$ and u is of base type (We assume w.l.o.g. that the names and variables in $\tilde{r}$ are not in D').* Note that in such a case $c \notin \tilde{r}$. Moreover, note this is the only possible combinaison of input and output since C' is an attacker evaluation context and A' is an honest extended process. Let us consider a variable z such that $z \notin fv(C'[A']) \cup bn(C'[A'])$. Hence $A' \xrightarrow{\nu z.eav(c,z)} \nu\tilde{r}.(P_1 \mid P_2\{^{u}/_{x}\} \mid P_3 \mid \{^{u}/_{z}\})$. But since $z \notin fv(C'[A']) \cup bn(C'[A'])$, we deduce that $B \equiv \nu\tilde{k}'.\nu z.\nu\tilde{r}.(Q_1\{^{u}/_{y}\} \mid Q_2 \mid P_1 \mid P_2\{^{u}/_{x}\} \mid P_3 \mid \{^{u}/_{z}\})$. Hence, by denoting $C''[\_] = \nu\tilde{k}'.\nu z.(Q_1\{^{z}/_{y}\} \mid Q_2 \mid \_)$ and $A'' = \nu\tilde{r}.(P_1 \mid P_2\{^{u}/_{x}\} \mid P_3 \mid \{^{u}/_{z}\})$, the result holds.

Case 9, rule C-OEAV, i.e. $A' = \nu\tilde{r}.(\mathsf{out}^{\mathsf{ho}}(c, d).P_1 \mid \mathsf{in}^{\mathsf{ho}}(c, x).P_2 \mid P_3)$, $D'' = \mathsf{eav}(c, y).Q_1 \mid Q_2)$, $B \equiv \nu\tilde{k}'.\nu\tilde{r}.(Q_1\{^d/_y\} \mid Q_2 \mid P_1 \mid P_2\{^d/_x\} \mid P_3 \mid \omega d)$ *and d is of channel type (We assume w.l.o.g. that the names and variables in $\tilde{r}$ are not in D').* We have to do a case analysis on d:

- Case $d \in \tilde{r}$: Let us denote $\nu\tilde{r} = \nu\tilde{r}'.\nu d$. In such a case $A' \xrightarrow{\nu d.eav(c,d)} \nu\tilde{r}'.(P_1 \mid P_2\{^u/_x\} \mid P_3)$. But we know that the names and variables in $\tilde{r}$ are not in D' hence $B'' \equiv \nu\tilde{k}'.\nu d.(Q_1\{^d/_y\} \mid Q_2 \mid \nu\tilde{r}'.(P_1 \mid P_2\{^d/_x\} \mid P_3))$. Therefore, by denoting $C''[\_] = \nu\tilde{k}'.\nu d.(Q_1\{^u/_y\} \mid Q_2 \mid \omega d \mid \_)$ and $A'' = \nu\tilde{r}'.(P_1 \mid P_2\{^d/_x\} \mid P_3)$, the result holds.
- Case $d \notin \tilde{r}$: In such a case $A' \xrightarrow{eav(c,u)} \nu\tilde{r}.(P_1 \mid P_2\{^u/_x\} \mid P_3)$ and so the result holds by denoting $C''[\_] = \nu\tilde{k}'.(Q_1\{^u/_y\} \mid Q_2 \mid \omega d \mid \_)$ and $A'' = \nu\tilde{r}.(P_1 \mid P_2\{^u/_x\} \mid P_3)$.

Note that in both case, $C''[\_]$ is indeed e-closing for A''.

We have proved that for all $s \in \{\mathsf{c}, \mathsf{p}, \mathsf{e}\}$, for all $C^s[A] \rightarrow_s^* B$, there exist an attacker evaluation context $C'[\_] = \nu\tilde{k}'.(D' \mid \_)$ with D' name-cleaned, an honest extended process A' and tr such that $C'[\_]$ is s-closing for A', $B \equiv C'[A']$ and $A \overset{\mathsf{tr}}{\Rightarrow}_s A'$. This property allows us to conclude the main proof. Indeed, consider $s \in \{\mathsf{c}, \mathsf{e}, \mathsf{p}\}$ and $C^s[\_]$ an attacker evaluation context such that $C^s[A] \Downarrow_c^{s,\mathsf{ho}}$. By definition, we deduce that $C^s[A] \rightarrow_s^* C[\mathsf{out}^{\mathsf{ho}}(c, t).P]$ for some evaluation context C that does not bind c, some t and some plain process P. By our property, we deduce that there exists an attacker evaluation context $C'[\_] = \nu\tilde{k}'.(D' \mid \_)$ with D' name-cleaned, an honest extended process A' and tr such that $C[\mathsf{out}^{\mathsf{ho}}(c, t).P] \equiv C'[A']$ and $A \overset{\mathsf{tr}}{\Rightarrow}_s A'$. More specifically, since $C'[\_]$ is an attacker evaluation context, $C[\mathsf{out}^{\mathsf{ho}}(c, t).P] \equiv C'[A']$ and C does not bind c, we deduce that $A' \equiv \nu\tilde{r}.(\mathsf{out}^{\mathsf{ho}}(c, t').P' \mid Q')$ for some t', P', Q', $\tilde{r}$ such that $c \notin \tilde{r}$. Therefore, if $t' \in \mathcal{C}h$ but $t' \notin \tilde{r}$ then $A' \xrightarrow{out(c,t')}_s A''$ for some A'' meaning that $A \xRightarrow{\mathsf{tr}.out(c,t')}_s A''$; else $A' \xrightarrow{\nu z.out(c,z)}_s A''$ for some A'' and some z fresh (z being either a base type variable or a channel), meaning that $A \xRightarrow{\mathsf{tr}.\nu z.out(c,z)}_s A''$. In both cases, we obtain that $A \overset{\mathsf{tr}'}{\Rightarrow} A''$, $out(c, t) \in \mathsf{tr}'$ and $c \notin bn(\mathsf{tr}')$ for some tr', A'' and t. It allows us to conclude that $A \Downarrow_c^s$. □

Appendix C. Proof of Theorem 1

We start by restating the a proposition from [15] that was used to prove that trace equivalence implies may equivalence in the classical semantics. In order to prove the proposition for the semantics private and eavesdrop, we will first write exactly the proof of from [15] for the classical semantics and then highlight what changes are required to obtain the proofs for the private and eavesdropping semantics.

Proposition 3. *Let $s \in \{\mathsf{c}, \mathsf{p}, \mathsf{e}\}$. Let A and B be two honest closed extended process with* $\mathrm{dom}(A) = \mathrm{dom}(B)$*, and $C[\_] = \nu\tilde{n}.(D \mid \_)$ be an attacker evaluation context s-closing for A. If $C[A] \rightarrow_s^* A''$ for some process A'', then there exist a closed extended process A', an attacker evaluation context $C' = \nu\tilde{n}'.(D' \mid \_)$ s-closing for A', and a trace* $\mathsf{tr} \in (\mathcal{A} \smallsetminus \{\tau\})^*$ *such that $A'' \equiv C'[A']$, $A \overset{\mathsf{tr}}{\Rightarrow}_s A'$, and for all closed extended process B', we have:*

$$\begin{array}{c} C[\_] \text{ is } s\text{-closing for } B \text{ and } B \overset{\mathsf{tr}}{\Rightarrow}_s B' \text{ and } \phi(B') \sim \phi(A') \\ \text{implies that} \\ C' \text{ is } s\text{-closing for } B' \text{ and } C[B] \rightarrow_s^* C'[B']. \end{array}$$

Proof. We first focus on the case $s = \mathsf{c}$. Let A and B be two extended processes with $\mathrm{dom}(A) = \mathrm{dom}(B)$ and $C[\_] = \nu\tilde{n}.(D \mid \_)$ be an evaluation context c-closing for A. Let A'' be such that $C[A] \rightarrow_s^* A''$. We prove the result by induction on the length ℓ of the derivation.

Base case $\ell = 0$: In such a case, we have that $A'' \equiv C[A]$. Let $A' = A$, $C' = C$ and $\mathsf{tr} = \epsilon$, we have that $A'' \equiv C'[A']$, and $A \overset{\mathsf{tr}}{\Rightarrow}_{\mathsf{c}} A'$. Let B' be a closed extended process such that $B \overset{\epsilon}{\Rightarrow}_{\mathsf{c}} B'$ and $\phi(B') \sim \phi(A')$ for some B'. Clearly, we have that $C[B] \rightarrow_{\mathsf{c}}^* C'[B']$ and $C'[\_]$ is c-closing for B' since $C' = C$ and $B \rightarrow_{\mathsf{c}}^* B'$.

Inductive case $\ell > 0$: In such a case, we have that there exists a closed extended process A_1 such that $C[A] \rightarrow_{\mathsf{c}}^* A_1$ with a derivation whose length is smaller than ℓ, and $A_1 \rightarrow_{\mathsf{c}} A''$. Thus, we can apply our induction hypothesis allowing us to deduce that there exist an extended process A'_1, an evaluation context $C'_1[\_] = \nu\tilde{n}'_1.(D'_1 \mid \_)$ c-closing for A'_1, and a trace $\mathsf{tr}_1 \in (\mathcal{A} \setminus \{\tau\})^*$ such that $A_1 \equiv C'_1[A'_1]$, $A \overset{\mathsf{tr}_1}{\Rightarrow}_{\mathsf{c}} A'_1$, and for all closed extended processes B'_1, we have that:

$$\begin{array}{c} C[\_] \text{ is c-closing for } B \text{ and } B \overset{\mathsf{tr}}{\Rightarrow}_s B'_1 \text{ and } \phi(B'_1) \sim \phi(A'_1) \\ \text{implies that} \\ C'_1[\_] \text{ is c-closing for } B'_1 \text{ and } C[B] \rightarrow_s^* C'_1[B'_1]. \end{array}$$

Since $A_1 \equiv C'_1[A'_1]$ and $A_1 \rightarrow_{\mathsf{c}} A''$, we have that $C'_1[A'_1] \rightarrow_{\mathsf{c}} A''$. (internal reduction is closed under structural equivalence). W.l.o.g., we can assume that D'_1 is name-cleaned, the bound names and variables in $C'_1[A'_1]$ are bound once and distinct from the free names. We do a case analysis on the rule involved in this reduction.

Case 1: internal reduction in A'_1, i.e. there exists A' such that $A'_1 \rightarrow_{\mathsf{c}} A'$ and $A'' \equiv C'_1[A']$. In such a case, we have that $C'_1[A'_1] \rightarrow_{\mathsf{c}} C'_1[A']$. Let $C'[\_] = C'_1[\_]$ and $\mathsf{tr} = \mathsf{tr}_1$. We have that $A'' \equiv C'_1[A'] = C'[A']$ and $A \overset{\mathsf{tr}_1}{\Rightarrow}_{\mathsf{c}} A'_1 \rightarrow_{\mathsf{c}} A'$, *i.e.* $A \overset{\mathsf{tr}}{\Rightarrow}_{\mathsf{c}} A'$. Lastly, let B' be a closed extended process such that $B \overset{\mathsf{tr}}{\Rightarrow}_{\mathsf{c}} B'$ and $\phi(B') \sim \phi(A')$. We have that $B \overset{\mathsf{tr}_1}{\Rightarrow}_{\mathsf{c}} B'$ and $\phi(B') \sim \phi(A'_1) \equiv \phi(A')$, and thus relying on our induction hypothesis, we obtain that $C'_1[\_]$ is c-closing for B' and $C[B] \rightarrow_{\mathsf{c}}^* C'_1[B']$. Since $C'_1[\_] = C'[\_]$, we conclude.

Case 2.a: rule THEN *in D'_1, i.e. $D'_1 =$ $\mathsf{if}\ u = v\ \mathsf{then}\ P_1\ \mathsf{else}\ P_2 \mid P_3$ and $A'' \equiv \nu\tilde{n}'_1.(P_1 \mid P_3 \mid A'_1)$.* In such a case, we have $C'_1[A'_1] \rightarrow_{\mathsf{c}} \nu\tilde{n}'_1.(P_1 \mid P_3 \mid A'_1)$. Let $A' = A'_1$, $C'[\_] = \nu\tilde{n}'_1.(P_1 \mid P_3 \mid \_)$ and $\mathsf{tr} = \mathsf{tr}_1$. We have that $A'' \equiv C'[A']$ and $A \overset{\mathsf{tr}}{\Rightarrow}_{\mathsf{c}} A'$. Lastly, let B' be a closed extended process such that $B \overset{\mathsf{tr}}{\Rightarrow}_{\mathsf{c}} B'$ and $\phi(B') \sim \phi(A')$. By renaming, we can assume that the bound names of B' are distinct from the free names of C'_1. Moreover, we know that C'_1 is c-closing for A'_1 meaning that $v(u, v) \subseteq \mathrm{dom}(\phi(A'_1))$. Furthermore, since the free names are distinct from bound names, we obtain that $fn(u, v) \cap bn(A'_1) = \emptyset$. But $\phi(A'_1) = \phi(A') \sim \phi(B')$ and $(u =_E v)\phi(A')$ hence we obtain $(u =_E v)\phi(B')$ meaning that $C'_1[B'] \rightarrow \nu\tilde{n}'_1.(P_1 \mid P_3 \mid B') = C'[B']$. By our inductive hypothesis, we also have that C'_1 is c-closing for B' and $C[B] \rightarrow_{\mathsf{c}}^* C'_1[B']$. Hence, we conclude that $C[B] \rightarrow_{\mathsf{c}}^* C'[B']$ and $C'[\_]$ is c-closing for B'.

Case 2.b: rule ELSE *in D'_1, i.e. $D'_1 =$ $\mathsf{if}\ u = v\ \mathsf{then}\ P_1\ \mathsf{else}\ P_2 \mid P_3$ and $A'' \equiv \nu\tilde{n}'_1.(P_2 \mid P_3 \mid A'_1)$.* Similar to case 2.a.

Case 3: rule COMM *in D'_1, i.e. $D'_1 = \mathsf{out}^{\mathsf{at}}(c, u).P_1 \mid \mathsf{in}^{\mathsf{at}}(c, x).P_2 \mid P_3$ and $A'' \equiv \nu\tilde{n}'_1.(P_1 \mid P_2\{^u/_x\} \mid P_3 \mid A'_1)$.* In such a case, we have $C'_1[A'_1] \rightarrow_{\mathsf{c}} \nu\tilde{n}'_1.(P_1 \mid P_2\{^u/_x\} \mid P_3 \mid A'_1)$. Let $A' = A'_1$, $C'[\_] = \nu\tilde{n}'_1.(P_1 \mid P_2\{^u/_x\} \mid P_3 \mid \_)$ and $\mathsf{tr} = \mathsf{tr}_1$. We have that $A'' \equiv C'[A']$ and $A \overset{\mathsf{tr}}{\Rightarrow}_{\mathsf{c}} A'$. Lastly, let B' be a closed extended process such that $B \overset{\mathsf{tr}}{\Rightarrow}_{\mathsf{c}} B'$ and $\phi(B') \sim \phi(A')$. Since $\phi(A') = \phi(A'_1)$ then by our inductive hypothesis, we obtain C'_1 is c-closing for B' and $C[B] \rightarrow_{\mathsf{c}}^{*} C'_1[B']$. But $C'_1[B'] \rightarrow_{\mathsf{c}} \nu\tilde{n}'_1.(P_1 \mid P_2\{^{u}/_{x}\} \mid P_3 \mid B') = C'[B']$ and so the result holds.

Case 4: rule COMM *between D'_1 (output) and A'_1 (input), i.e. $D'_1 = \mathsf{out}^{\mathsf{at}}(c, M).P_1 \mid P_2$, $A'_1 = \nu\tilde{r}.(\mathsf{in}^{\mathsf{ho}}(c, x).Q_1 \mid Q_2)$ and $A'' \equiv \nu\tilde{n}'_1.\nu\tilde{r}.(P_1 \mid P_2 \mid Q_1 \mid Q_2)$ (recall that we assume that bound names and variables are distinct from free names and variables and are only bound once).* Note that in such a case, $c \notin \tilde{r}$. Hence $A'_1 \xrightarrow{in(c,M)} \nu\tilde{r}.(Q_1\{^{M}/_{x}\} \mid Q_2)$. Moreover, since $\tilde{r}$ are not in P_1, P_2, we have $A'' \equiv \nu\tilde{n}'_1.(P_1 \mid P_2 \mid \nu\tilde{r}.(Q_1\{^{M}/_{x}\} \mid Q_2))$. Let $A' = \nu\tilde{r}.(Q_1\{^{M}/_{x}\} \mid Q_2)$, $C'[\_] = \nu\tilde{n}'_1.(P_1 \mid P_2 \mid \_)$ and $\mathsf{tr} = \mathsf{tr}_1.in(c, M)$. We have that $A'' \equiv C'[A']$ and $A \overset{\mathsf{tr}}{\Rightarrow}_{\mathsf{c}} A'$. Lastly let B' be a closed extended process such that $B \overset{\mathsf{tr}}{\Rightarrow}_{\mathsf{c}} B'$ and $\phi(B') \sim \phi(A')$. We have that there exists B'_1 such that $B \overset{\mathsf{tr}_1}{\Rightarrow}_{\mathsf{c}} B'_1 \xrightarrow{in(c,M)}_{\mathsf{c}} B'_2 \rightarrow_{\mathsf{c}}^{*} B'$. By renaming, we can assume that the bound names of B'_1 are distinct from the names of C'_1 and are bound only once. Since $\phi(B') \sim \phi(A')$, we have also that $\phi(B'_1) \sim \phi(A'_1)$. Thus, we can apply our induction hypothesis on B'_1. This allows us to deduce that $C[B] \rightarrow_{\mathsf{c}}^{*} C'_1[B'_1]$ and C'_1 is c-closing for B'_1. In order to conclude, it remains to show that $C'_1[B'_1] \rightarrow_{\mathsf{c}} C'[B'_2]$ and $C'[\_]$ is c-closing for B'_2 (since $C'[\_]$ is c-closing for B'_2 and $B'_2 \rightarrow_{\mathsf{c}}^{*} B'$ implies $C'[B'_2] \rightarrow_{\mathsf{c}}^{*} C'[B']$ and C' is c-closing for B').

We have seen that $B'_1 \xrightarrow{in(c,M)} B'$. Hence, we know that $B'_1 = \nu\tilde{r}'.(\mathsf{in}^{\mathsf{ho}}(c, x).Q'_1 \mid Q'_2)$ for some $\tilde{r}'$, Q'_1, Q'_2 and $B'_2 \equiv \nu\tilde{r}'.(Q'_1\{^{M}/_{x}\} \mid Q'_2)$. But since we assumed that the bound names of B'_1 are distinct from the names of C'_1 and are bound only once, we obtain that $C'_1[B'_1] \equiv \nu\tilde{n}'_1.\nu\tilde{r}'.(\mathsf{out}^{\mathsf{at}}(c, M).P_1 \mid P_2 \mid \mathsf{in}^{\mathsf{ho}}(c, x).Q'_1 \mid Q'_2)$. Hence $C'_1[B'_1] \rightarrow_{\mathsf{c}} \nu\tilde{n}'_1.\nu\tilde{r}'.(P_1 \mid P_2 \mid Q'_1\{^{M}/_{x}\} \mid Q'_2) \equiv C'[\nu\tilde{r}'.(Q'_1\{^{M}/_{x}\} \mid Q'_2)] \equiv C'[B'_2]$. Notice that $C'[\_]$ is c-closing for B'_2 since $fv(C'_1[B'_1]) = \emptyset$.

Case 5: rule COMM *between C'_1 (input) and A'_1 (output), i.e. $D'_1 = \mathsf{in}^{\mathsf{at}}(c, x).P_1 \mid P_2$, $A'_1 = \nu\tilde{r}.(\mathsf{out}^{\mathsf{ho}}(c, M).Q_1 \mid Q_2)$ and $A'' \equiv \nu\tilde{n}'_1.\nu\tilde{r}.(P_1\{^{M}/_{x}\} \mid P_2 \mid Q_1 \mid Q_2)$ (recall that we assume that bound names and variables are distinct from free names and variables and are only bound once).* Note that in such a case, $c \notin \tilde{r}$. We do a case analysis on M.

- Case 5.a, $M \in \mathcal{C}h \cap \tilde{r}$: Let us denote $\nu\tilde{r} = \nu\tilde{r}'.\nu M$. Thus $A'_1 \xrightarrow{\nu M.out(c,M)}_{\mathsf{c}} \nu\tilde{r}'.(Q_1 \mid Q_2)$. Hence, since the names and variables in $\tilde{r}$ are not in D'_1, we obtain that $A'' \equiv \nu\tilde{n}'_1.\nu M.(P_1\{^{M}/_{x}\} \mid P_2 \mid \nu\tilde{r}'.(Q_1 \mid Q_2))$. Let $A' = \nu\tilde{r}'.(Q_1 \mid Q_2)$, $C'[\_] = \nu\tilde{n}'_1.\nu M.(P_1\{^{M}/_{x}\} \mid P_2 \mid \_)$ and $\mathsf{tr} = \mathsf{tr}_1.\nu M.out(c, M)$. We have that $A'' \equiv C'[A']$ and $A \overset{\mathsf{tr}}{\Rightarrow}_{\mathsf{c}} A'$. Lastly let B' be a closed extended process such that $B \overset{\mathsf{tr}}{\Rightarrow}_{\mathsf{c}} B'$ and $\phi(B') \sim \phi(A')$. We have that there exists B'_1 such that $B \overset{\mathsf{tr}_1}{\Rightarrow}_{\mathsf{c}} B'_1 \xrightarrow{\nu M.out(c,M)}_{\mathsf{c}} B'_2 \rightarrow_{\mathsf{c}}^{*} B'$. By renaming, we can assume that the bound names of B'_1 are distinct from the names of C'_1 and are bound only once. Since $\phi(B') \sim \phi(A')$, we have also that $\phi(B'_1) \sim \phi(A'_1)$. Thus, we can apply our induction hypothesis on B'_1. This allows us to deduce that $C[B] \rightarrow_{\mathsf{c}}^{*} C'_1[B'_1]$ and $C'_1[\_]$ is c-closing for B'_1. In order to conclude, it remains to show that $C'_1[B'_1] \rightarrow_{\mathsf{c}} C'[B'_2]$ (since $fv(C'_1[B'_1])$ and since $B'_2 \rightarrow_{\mathsf{c}}^{*} B'$ implies $C'[B'_2] \rightarrow_{\mathsf{c}}^{*} C'[B']$).
  We have seen that $B'_1 \xrightarrow{\nu M.out(c,M)} B'_2$. Hence, $B'_1 = \nu\tilde{m}.\nu M.(\mathsf{out}^{\mathsf{ho}}(c, M).Q'_1 \mid Q'_2)$ for some $\tilde{m}$, Q'_1, Q'_2 and $B'_2 \equiv \nu\tilde{m}.(Q'_1 \mid Q'_2)$. But since we assumed that the bound names of B'_1 are distinct from the names of C'_1 and are bound only once, we obtain that $C'_1[B'_1] \equiv \nu\tilde{n}'_1.\nu\tilde{m}.\nu M.(\mathsf{in}^{\mathsf{at}}(c, x).P_1 \mid P_2 \mid \mathsf{out}^{\mathsf{ho}}(c, M).Q'_1 \mid Q'_2)$. Hence $C'_1[B'_1] \rightarrow_{\mathsf{c}} \nu\tilde{n}'_1.\nu\tilde{m}.\nu M.(P_1\{^{M}/x\} \mid P_2 \mid Q'_1 \mid Q'_2) \equiv C'[\nu\tilde{m}.(Q'_1 \mid Q'_2)] \equiv C'[B'_2]$.
- Case 5.b, $M \in \mathcal{C}h$ but $M \notin \tilde{r}$: Thus $A'_1 \xrightarrow{out(c,M)}_{\mathsf{c}} \nu\tilde{r}.(Q_1 \mid Q_2)$. Hence, since the names and variables in $\tilde{r}$ are not in D'_1, we obtain that $A'' \equiv \nu\tilde{n}'_1.(P_1\{^{M}/_{x}\} \mid P_2 \mid \nu\tilde{r}.(Q_1 \mid Q_2))$. Let $A' = \nu\tilde{r}.(Q_1 \mid Q_2)$, $C'[\_] = \nu\tilde{n}'_1.(P_1\{^{M}/_{x}\} \mid P_2 \mid \_)$ and $\mathsf{tr} = \mathsf{tr}_1.out(c, M)$. We have that $A'' \equiv C'[A']$

and $A \overset{\mathsf{tr}}{\Rightarrow}_{\mathsf{c}} A'$. Lastly let B' be a closed extended process such that $B \overset{\mathsf{tr}}{\Rightarrow}_{\mathsf{c}} B'$ and $\phi(B') \sim \phi(A')$. We have that there exists B'_1 such that $B \overset{\mathsf{tr}_1}{\Rightarrow}_{\mathsf{c}} B'_1 \xrightarrow{out(c,M)}_{\mathsf{c}} B'_2 \rightarrow^*_{\mathsf{c}} B'$. By renaming, we can assume that the bound names of B'_1 are distinct from the names of C'_1 and are bound only once. Since $\phi(B') \sim \phi(A')$, we have also that $\phi(B'_1) \sim \phi(A'_1)$. Thus, we can apply our induction hypothesis on B'_1. This allows us to deduce that $C[B] \rightarrow^*_{\mathsf{c}} C'_1[B'_1]$ and $C'_1[\_]$ is c-closing for B'_1. In order to conclude, it remains to show that $C'_1[B'_1] \rightarrow_{\mathsf{c}} C'[B']$ (since $fv(C'_1[B'_1])$ and since $B'_2 \rightarrow^*_{\mathsf{c}} B'$ implies $C'[B'_2] \rightarrow^*_{\mathsf{c}} C'[B']$).

We have seen that $B'_1 \xrightarrow{out(c,M)} B'_2$. Hence, $B'_1 = \nu\tilde{m}.(\mathsf{out}^{\mathsf{ho}}(c, M).Q'_1 \mid Q'_2)$ for some $\tilde{m}$, Q'_1, Q'_2 such that $M \notin \tilde{m}$ and $B'_2 \equiv \nu\tilde{m}.(Q'_1 \mid Q'_2)$. But since we assumed that the bound names of B'_1 are distinct from the names of C'_1 and are bound only once, we obtain that $C'_1[B'_1] \equiv \nu\tilde{n}'_1.\nu\tilde{m}.(\mathsf{in}^{\mathsf{at}}(c, x).P_1 \mid P_2 \mid \mathsf{out}^{\mathsf{ho}}(c, M).Q'_1 \mid Q'_2)$. Hence $C'_1[B'_1] \rightarrow_{\mathsf{c}} \nu\tilde{n}'_1.\nu\tilde{m}.(P_1\{^M/x\} \mid P_2 \mid Q'_1 \mid Q'_2) \equiv C'[\nu\tilde{m}.(Q'_1 \mid Q'_2)] \equiv C'[B'_2]$.

- Case 5.c, $M \notin \mathcal{C}h$: Consider y a fresh variable. Thus $A'_1 \xrightarrow{\nu y.out(c,y)}_{\mathsf{c}} \nu\tilde{r}.(Q_1 \mid Q_2 \mid \{^M/_y\})$. Hence, since the names and variables in $\tilde{r}$ are not in D'_1 and since y is fresh, we obtain that $A'' \equiv \nu\tilde{n}'_1.\nu y.\nu\tilde{r}.(P_1\{^y/_x\} \mid P_2 \mid Q_1 \mid Q_2 \mid \{^M/_y\}) \equiv \nu\tilde{n}'_1.\nu y.(P_1\{^y/_x\} \mid P_2 \mid \nu\tilde{r}.(Q_1 \mid Q_2 \mid \{^M/_y\}))$. Let $A' = \nu\tilde{r}.(Q_1 \mid Q_2 \mid \{^M/_y\})$, $C'[\_] = \nu\tilde{n}'_1.\nu y.(P_1\{^y/_x\} \mid P_2 \mid \_)$ and $\mathsf{tr} = \mathsf{tr}_1.\nu y.out(c, y)$. We have that $A'' \equiv C'[A']$ and $A \overset{\mathsf{tr}}{\Rightarrow}_{\mathsf{c}} A'$. Lastly let B' be a closed extended process such that $B \overset{\mathsf{tr}}{\Rightarrow}_{\mathsf{c}} B'$ and $\phi(B') \sim \phi(A')$. We have that there exists B'_1 such that $B \overset{\mathsf{tr}_1}{\Rightarrow}_{\mathsf{c}} B'_1 \xrightarrow{\nu y.out(c,y)}_{\mathsf{c}} B'_2 \rightarrow^*_{\mathsf{c}} B'$. By renaming, we can assume that the bound names of B'_1 are distinct from the names of C'_1 and are bound only once. Since $\phi(B') \sim \phi(A')$, we deduce that $\mathrm{dom}(B'_1) = \mathrm{dom}(A'_1)$ and $\phi(B'_1) \sim \phi(A'_1)$. Thus, we can apply our induction hypothesis on B'_1. This allows us to deduce that $C[B] \rightarrow^*_{\mathsf{c}} C'_1[B'_1]$ and $C'_1[\_]$ is c-closing for B'_1. In order to conclude, it remains to show that $C'_1[B'_1] \rightarrow_{\mathsf{c}} C'[B']$ (since $fv(C'_1[B'_1])$ and since $B'_2 \rightarrow^*_{\mathsf{c}} B'$ implies $C'[B'_2] \rightarrow^*_{\mathsf{c}} C'[B']$).

  We have seen that $B'_1 \xrightarrow{\nu y.out(c,y)} B'_2$. Hence, $B'_1 = \nu\tilde{m}.(\mathsf{out}^{\mathsf{ho}}(c, N).Q'_1 \mid Q'_2)$ for some $\tilde{m}$, $N \notin \mathcal{C}h$, Q'_1, Q'_2 and $B'_2 \equiv \nu\tilde{m}.(Q'_1 \mid Q'_2 \mid \{^N/_y\})$. But since we assumed that the bound names of B'_1 are distinct from the names of C'_1 and are bound only once, we obtain that $C'_1[B'_1] \equiv \nu\tilde{n}'_1.\nu\tilde{m}.(\mathsf{in}^{\mathsf{at}}(c, x).P_1 \mid P_2 \mid \mathsf{out}^{\mathsf{ho}}(c, N).Q'_1 \mid Q'_2)$. Moreover, since y is fresh, we obtain that $\nu\tilde{n}'_1.\nu y.\nu\tilde{m}.(\mathsf{in}^{\mathsf{at}}(c, x).P_1 \mid P_2 \mid \mathsf{out}^{\mathsf{ho}}(c, y).Q'_1 \mid Q'_2 \mid \{^N/_y\})$. Hence $C'_1[B'_1] \rightarrow_{\mathsf{c}} \nu\tilde{n}'_1.\nu y.\nu\tilde{m}.(P_1\{^y/x\} \mid P_2 \mid Q'_1 \mid Q'_2 \mid \{^N/_y\}) \equiv C'[\nu\tilde{m}.(Q'_1 \mid Q'_2 \mid \{^N/_y\})] \equiv C'[B'_2]$.

This conclude the proof of the proposition for $s = \mathsf{c}$. Therefore, it remains to take care of the cases $s = \mathsf{p}$ and $s = \mathsf{e}$. Let us focus first on the case $s = \mathsf{p}$. The proof is in fact very similar to the classical semantics. Considering that the differences between the classical semantics and the private semantics are on the internal communication, we only need the rules that are not already covered in the classical proof. Notice that the rule C-ENV correspond to either Case 3 or Case 4 when M is of base type or Case 5.c. Moreover, the rules THEN and ELSE are already covered either Case 1 or 2.a or 2.b. Hence it remains the case of the rules C-PRIV and C-OPEN.

Case 6, rule C-OPEN *between C'_1 (input) and A'_1 (output), i.e.* $D'_1 = \mathsf{in}^{\theta}(c, x).P_1 \mid P_2$, $A'_1 = \nu\tilde{r}.(\mathsf{out}^{\mathsf{ho}}(c, d).Q_1 \mid Q_2)$ *and* $A'' \equiv \nu\tilde{n}'_1.\nu\tilde{r}.(P_1\{^d/_x\} \mid P_2 \mid Q_1 \mid Q_2 \mid \omega d)$. Lets us do a case analysis on whether (6.a) $d \in \tilde{r}$ or (6.b) $d \notin \tilde{r}$. Note that Case (6.a) is in fact almost identical to Case (5.a) and that the result holds with $C'[\_] = \nu\tilde{n}'_1.\nu d.(P_1\{^d/_x\} \mid P_2 \mid \omega d \mid \_)$, $A' = \nu\tilde{r}'.(Q_1 \mid Q_2)$ and $\mathsf{tr} = \mathsf{tr}.\nu d$ with $\nu\tilde{r} = \nu\tilde{r}'.\nu d$. Furthermore, note that Case (6.b) is also very similar to Case (5.b) and that the result holds with $C'[\_] = \nu\tilde{n}'_1.(P_1\{^d/_x\} \mid P_2 \mid \omega d \mid \_)$ and $A'' = \nu\tilde{r}.(Q_1 \mid Q_2)$. Notice that in both cases $C'[\_]$ is p-closing for A' and B' since ωd was added to $C'[\_]$.

Case 7, rule C-OPEN *between* A'_1 *(input) and* C'_1 *(output), i.e.* $D'_1 = \mathsf{out}^{\theta}(c,d).P_1 \mid P_2$, $A'_1 = \nu\tilde{r}.(\mathsf{in}^{\mathsf{ho}}(c,x).Q_1 \mid Q_2)$ *and* $A'' \equiv \nu\tilde{n}'_1.\nu\tilde{r}.(P_1 \mid P_2 \mid Q_1\{^d/_x\} \mid Q_2 \mid \omega d)$. This case if very similar to Case 4 when M is of channel type and the result holds with $C'[\_] = \nu\tilde{n}'_1.(P_1 \mid P_2 \mid \omega d \mid \_)$ and $A' = \nu\tilde{r}.(Q_1\{^d/_x\} \mid Q_2)$. Notice that in both cases $C'[\_]$ is p-closing for A' and B' since ωd was added to $C'[\_]$.

Case 8, rule C-PRIV *with a communication on a channel* c. Notice that this rule is in fact partially covered by the beginning of the proof. Indeed, Case 1 and 3 cover the cases where c is not in $\tilde{n}'_1$. Therefore, we only need to focus on the case where the private channel is in $\tilde{n}'_1$, *i.e.* $\nu\tilde{n}'_1 = \nu\tilde{n}''_1.\nu c$ for some $\tilde{n}''_1$. We know that $C'_1[\_]$ is p-closing for A'_1. Hence since c is a channel bound in $C'_1[\_]$ whose scope includes _, we deduce that if $c \in fn(A_1)$ then ωc is also in the scope of c. But according to the definition of the rule, we know that ωc is not in the scope of νc. Moreover, if the output or input is done by A'_1 then it would implies that $c \in fn(A_1)$. Thus, this allows us to deduce that this both output and input are tagged with at, meaning that there exists D''_1 such that $\nu c.(D'_1 \mid A'_1) \xrightarrow{\tau}_{\mathsf{p}} \nu c.(D''_1 \mid A'_1)$ and $A'' \equiv \nu\tilde{n}''_1.\nu c.(D''_1 \mid A'_1)$. In such a case, by denoting $C'[\_] = \nu\tilde{n}''_1.\nu c.(D''_1 \mid \_)$, $A' = A'_1$ and $\mathsf{tr} = \mathsf{tr}_1$, we obtain $A'' \equiv C'[A']$ and $A \overset{\mathsf{tr}_1}{\Longrightarrow}_{\mathsf{p}} A'$. Lastly let B' be a closed extended process such that $B \overset{\mathsf{tr}}{\Longrightarrow}_{\mathsf{p}} B'$ and $\phi(B') \sim \phi(A')$. By our inductive hypothesis, we know that $C[B] \rightarrow^{*}_{\mathsf{p}} C'_1[B']$. But $C'_1[B'] = \nu\tilde{n}''_1.\nu c.(D'_1 \mid B') \rightarrow_{\mathsf{p}} \nu c.(D''_1 \mid B') \equiv C'[B']$. Hence the result holds.

We have concluded the proof of the property for $s = \mathsf{p}$ hence it remains the case $s = \mathsf{e}$. Once, again several cases are already covered since $\xrightarrow{\ell}_{\mathsf{p}} \subset \xrightarrow{\ell}_{\mathsf{e}}$. Hence we only need to focus on the cases of the rules C-EAV and C-OEAV:

Case 8, rule C-EAV*,* i.e. $A'_1 = \nu\tilde{r}.(\mathsf{out}^{\mathsf{ho}}(c,u).Q_1 \mid \mathsf{in}^{\mathsf{ho}}(c,x).Q_2 \mid Q_3)$*,* $D'_1 = \mathsf{eav}(c,y).P_1 \mid P_2)$*,* $A'' \equiv \nu\tilde{n}'_1.\nu\tilde{r}.(P_1\{^u/_y\} \mid P_2 \mid Q_1 \mid Q_2\{^u/_x\} \mid Q_3)$ *and* u *is of base type (We assume w.l.o.g. that the names and variables in* $\tilde{r}$ *are not in* D'_1*).* Note that in such a case $c \notin \tilde{r}$. Moreover, note this is the only possible combinaison of input and output since C'_1 is an attacker evaluation context and A'_1 is an honest extended process. Let us consider a fresh variable z. Hence $A'_1 \xrightarrow{\nu z.eav(c,z)} \nu\tilde{r}.(Q_1 \mid Q_2\{^u/_x\} \mid Q_3 \mid \{^u/_z\})$. But since z is fresh, we deduce that $A'' \equiv \nu\tilde{n}'_1.\nu z.\nu\tilde{r}.(P_1\{^u/_y\} \mid P_2 \mid Q_1 \mid Q_2\{^u/_x\} \mid Q_3 \mid \{^u/_z\})$. Let $C'[\_] = \nu\tilde{n}'_1.\nu z.(P_1\{^z/_y\} \mid P_2 \mid \_)$, $A' = \nu\tilde{r}.(Q_1 \mid Q_2\{^u/_x\} \mid Q_3 \mid \{^u/_z\})$ and $\mathsf{tr} = \mathsf{tr}_1.\nu z.eav(c,z)$. We have $A'' \equiv C'[A']$ and $A \overset{\mathsf{tr}}{\Longrightarrow}_{\mathsf{p}} A'$. Let B' be a closed extended process such that $B \overset{\mathsf{tr}}{\Longrightarrow} B'$ and $\phi(B') \sim \phi(A')$. We have that there exists B'_1 such that $B \overset{\mathsf{tr}_1}{\Longrightarrow}_{\mathsf{e}} B'_1 \xrightarrow{\nu z.eav(c,z)}_{\mathsf{e}} B'_2 \rightarrow^{*}_{\mathsf{e}} B'$. By renaming, we can assume that the bound names of B'_1 are distinct from the names of C'_1 and are bound only once. Since $\phi(B') \sim \phi(A')$, we deduce that $\mathrm{dom}(B'_1) = \mathrm{dom}(A'_1)$ and $\phi(B'_1) \sim \phi(A'_1)$. Thus, we can apply our induction hypothesis on B'_1. This allows us to deduce that $C[B] \rightarrow^{*}_{\mathsf{e}} C'_1[B'_1]$ and $C'_1[\_]$ is e-closing for B'_1. In order to conclude, it remains to show that $C'_1[B'_1] \rightarrow_{\mathsf{e}} C'[B'_2]$ and $C'[\_]$ is p-closing for B'_2 (since $C'[\_]$ is e-closing for B'_2 and $B'_2 \rightarrow^{*}_{\mathsf{e}} B'$ implies $C'[B'_2] \rightarrow^{*}_{\mathsf{e}} C'[B']$ and $C'[\_]$ is p-closing for B').

We have seen that $B'_1 \xrightarrow{\nu z.eav(c,z)}_{\mathsf{e}} B'_2$. Hence, $B'_1 = \nu\tilde{m}.(\mathsf{out}^{\mathsf{ho}}(c,N).Q'_1 \mid \mathsf{in}^{\cdot}(c,x)Q'_2 \mid Q'_3)$ for some $\tilde{m}$, N is of base type, Q'_1, Q'_2, Q'_3 and $B'_2 \equiv \nu\tilde{m}.(Q'_1 \mid Q'_2\{^N/_x\} \mid Q'_3 \mid \{^N/_y\})$. But since we assumed that the bound names of B'_1 are distinct from the names of C'_1 and are bound only once, we obtain that $C'_1[B'_1] \equiv \nu\tilde{n}'_1.\nu\tilde{m}.(\mathsf{eav}(c,y).P_1 \mid P_2 \mid \mathsf{out}^{\mathsf{ho}}(c,N).Q'_1 \mid \mathsf{in}^{\cdot}(c,x)Q'_2 \mid Q'_3)$. Moreover, since z is fresh, we obtain that $\nu\tilde{n}'_1.\nu z.\nu\tilde{m}.(\mathsf{eav}(c,y).P_1 \mid P_2 \mid \mathsf{out}^{\mathsf{ho}}(c,z).Q'_1 \mid \mathsf{in}^{\cdot}(c,x)Q'_2 \mid Q'_3 \mid \{^N/_z\})$. Hence $C'_1[B'_1] \rightarrow_{\mathsf{e}} \nu\tilde{n}'_1.\nu y.\nu\tilde{m}.(P_1\{^z/_y\} \mid P_2 \mid Q'_1 \mid Q'_2\{^N/_x\} \mid Q'_3 \mid \{^N/_z\}) \equiv C'[\nu\tilde{m}.(Q'_1 \mid Q'_2\{^N/_x\} \mid Q'_3 \mid \{^N/_z\})] \equiv C'[B'_2]$. Note that since the rule is focused on base type terms, we directly have that $C'[\_]$ is e-closing for B'_2.

Case 9, rule C-OEAV, i.e. $A'_1 = \nu\tilde{r}.(\mathsf{out}^{\mathsf{ho}}(c,d).Q_1 \mid \mathsf{in}^{\mathsf{ho}}(c,x).Q_2 \mid Q_3)$*,* $D'_1 = \mathsf{eav}(c,y).P_1 \mid P_2)$*,* $A'' \equiv \nu\tilde{n}'_1.\nu\tilde{r}.(P_1\{^{d}/_{y}\} \mid P_2 \mid Q_1 \mid Q_2\{^{d}/_{x}\} \mid Q_3 \mid \omega d)$ *and d is of channel type (We assume w.l.o.g. that the names and variables in* $\tilde{r}$ *are not in* D'*).* We have to do a case analysis on d:

- Case $d \in \tilde{r}$: Let us denote $\nu\tilde{r} = \nu\tilde{r'}.\nu d$. In such a case $A'_1 \xrightarrow{\nu d.eav(c,d)} \nu\tilde{r}'.(Q_1 \mid Q_2\{^{d}/_{x}\} \mid Q_3)$. But we know that the names and variables in $\tilde{r}$ are not in D'_1 hence $A'' \equiv \nu\tilde{n}'_1.\nu d.(P_1\{^{d}/_{y}\} \mid P_2 \mid \omega d \mid \nu\tilde{r}'.(Q_1 \mid Q_2\{^{d}/_{x}\} \mid Q_3))$. Let $C'[\_] = \nu\tilde{k}'.\nu d.(Q_1\{^{d}/_{y}\} \mid Q_2 \mid \omega d \mid \_)$, $A' = \nu\tilde{r}'.(Q_1 \mid Q_2\{^{d}/_{x}\} \mid Q_3)$ and $\mathsf{tr} = \mathsf{tr}_1.\nu d.eav(c,d)$. We have $A'' \equiv C'[A']$ and $A \overset{\mathsf{tr}}{\Rightarrow}_{\mathsf{e}} A'$. Let B' be a closed extended process such that $B \overset{\mathsf{tr}}{\Rightarrow}_{\mathsf{e}} B'$ and $\phi(B') \sim \phi(A')$. We have that there exists B'_1 such that $B \overset{\mathsf{tr}_1}{\Rightarrow}_{\mathsf{e}} B'_1 \xrightarrow{\nu d.eav(c,d)}_{\mathsf{e}} B'_2 \rightarrow^{*}_{\mathsf{e}} B'$. By renaming, we can assume that the bound names of B'_1 are distinct from the names of C'_1 and are bound only once. Since $\phi(B') \sim \phi(A')$, we deduce that $\mathrm{dom}(B'_1) = \mathrm{dom}(A'_1)$ and $\phi(B'_1) \sim \phi(A'_1)$. Thus, we can apply our induction hypothesis on B'_1. This allows us to deduce that $C[B] \rightarrow^{*}_{\mathsf{e}} C'_1[B'_1]$ and $C'_1[\_]$ is e-closing for B'_1. In order to conclude, it remains to show that $C'_1[B'_1] \rightarrow_{\mathsf{e}} C'[B'_2]$ and $C'[\_]$ is e-closing for B'_2 (since $C'[\_]$ is e-closing for B'_2 and $B'_2 \rightarrow^{*}_{\mathsf{e}} B'$ implies $C'[B'_2] \rightarrow^{*}_{\mathsf{e}} C'[B']$ and $C'[\_]$ is e-closing for B').

  We have seen that $B'_1 \xrightarrow{\nu d.eav(c,d)}_{\mathsf{e}} B'_2$. Hence, $B'_1 = \nu\tilde{m}.\nu d.(\mathsf{out}^{\mathsf{ho}}(c,d).Q'_1 \mid \mathsf{in}^{\cdot}(c,x)Q'_2 \mid Q'_3)$ for some $\tilde{m}$, Q'_1, Q'_2, Q'_3 and $B'_2 \equiv \nu\tilde{m}.(Q'_1 \mid Q'_2\{^{d}/_{x}\} \mid Q'_3)$. But since we assumed that the bound names of B'_1 are distinct from the names of C'_1 and are bound only once, we obtain that $C'_1[B'_1] \equiv \nu\tilde{n}'_1.\nu\tilde{m}.\nu d.(\mathsf{eav}(c,y).P_1 \mid P_2 \mid \mathsf{out}^{\mathsf{ho}}(c,d).Q'_1 \mid \mathsf{in}^{\cdot}(c,x)Q'_2 \mid Q'_3)$. Hence $C'_1[B'_1] \rightarrow_{\mathsf{e}} \nu\tilde{n}'_1.\nu\tilde{m}.\nu d.(P_1\{^{d}/_{y}\} \mid P_2 \mid Q'_1 \mid Q'_2\{^{d}/_{x}\} \mid Q'_3 \mid \omega d) \equiv C'[\nu\tilde{m}.(Q'_1 \mid Q'_2\{^{d}/_{x}\} \mid Q'_3)] \equiv C'[B'_2]$. Note that d is possible a new free channel of B'_2. However, since we have ωd in C', we ensure that C' is e-closing for B'_2

- Case $d \notin \tilde{r}$: In such a case $A'_1 \xrightarrow{eav(c,d)} \nu\tilde{r}.(Q_1 \mid Q_2\{^{d}/_{x}\} \mid Q_3)$. But we know that the names and variables in $\tilde{r}$ are not in D'_1 hence $A'' \equiv \nu\tilde{n}'_1.(P_1\{^{d}/_{y}\} \mid P_2 \mid \omega d \mid \nu\tilde{r}.(Q_1 \mid Q_2\{^{d}/_{x}\} \mid Q_3))$. Let $C'[\_] = \nu\tilde{k}'.(Q_1\{^{d}/_{y}\} \mid Q_2 \mid \omega d \mid \_)$, $A' = \nu\tilde{r}.(Q_1 \mid Q_2\{^{d}/_{x}\} \mid Q_3)$ and $\mathsf{tr} = \mathsf{tr}_1.eav(c,d)$. We have $A'' \equiv C'[A']$ and $A \overset{\mathsf{tr}}{\Rightarrow}_{\mathsf{e}} A'$. Let B' be a closed extended process such that $B \overset{\mathsf{tr}}{\Rightarrow}_{\mathsf{e}} B'$ and $\phi(B') \sim \phi(A')$. We have that there exists B'_1 such that $B \overset{\mathsf{tr}_1}{\Rightarrow}_{\mathsf{e}} B'_1 \xrightarrow{eav(c,d)}_{\mathsf{e}} B'_2 \rightarrow^{*}_{\mathsf{e}} B'$. By renaming, we can assume that the bound names of B'_1 are distinct from the names of C'_1 and are bound only once. Since $\phi(B') \sim \phi(A')$, we deduce that $\mathrm{dom}(B'_1) = \mathrm{dom}(A'_1)$ and $\phi(B'_1) \sim \phi(A'_1)$. Thus, we can apply our induction hypothesis on B'_1. This allows us to deduce that $C[B] \rightarrow^{*}_{\mathsf{e}} C'_1[B'_1]$ and $C'_1[\_]$ is e-closing for B'_1. In order to conclude, it remains to show that $C'_1[B'_1] \rightarrow_{\mathsf{e}} C'[B'_2]$ and $C'[\_]$ is e-closing for B'_2 (since $C'[\_]$ is e-closing for B'_2 and $B'_2 \rightarrow^{*}_{\mathsf{e}} B'$ implies $C'[B'_2] \rightarrow^{*}_{\mathsf{e}} C'[B']$ and $C'[\_]$ is e-closing for B').

  We have seen that $B'_1 \xrightarrow{eav(c,d)}_{\mathsf{e}} B'_2$. Hence, $B'_1 = \nu\tilde{m}.(\mathsf{out}^{\mathsf{ho}}(c,d).Q'_1 \mid \mathsf{in}^{\cdot}(c,x)Q'_2 \mid Q'_3)$ with $d \notin \tilde{m}$ for some $\tilde{m}$, Q'_1, Q'_2, Q'_3 and $B'_2 \equiv \nu\tilde{m}.(Q'_1 \mid Q'_2\{^{d}/_{x}\} \mid Q'_3)$. But since we assumed that the bound names of B'_1 are distinct from the names of C'_1 and are bound only once, we obtain that $C'_1[B'_1] \equiv \nu\tilde{n}'_1.\nu\tilde{m}.(\mathsf{eav}(c,y).P_1 \mid P_2 \mid \mathsf{out}^{\mathsf{ho}}(c,d).Q'_1 \mid \mathsf{in}^{\cdot}(c,x)Q'_2 \mid Q'_3)$. Hence $C'_1[B'_1] \rightarrow_{\mathsf{e}} \nu\tilde{n}'_1.\nu\tilde{m}.(P_1\{^{d}/_{y}\} \mid P_2 \mid Q'_1 \mid Q'_2\{^{d}/_{x}\} \mid Q'_3 \mid \omega d) \equiv C'[\nu\tilde{m}.(Q'_1 \mid Q'_2\{^{d}/_{x}\} \mid Q'_3)] \equiv C'[B'_2]$. Note that d is possible a new free channel of B'_2 and b could be bound in $\tilde{n}'_1$. However, since we have ωd in C', we ensure that C' is e-closing for B'_2. □

Lemma 10. *Let A and B be two closed extended processes such that $A \approx_t^s B$. Let u be a name that occurs in $fn(A) \cup fn(B)$ and not in $bn(A) \cup bn(B)$, and u' be a fresh name. For all $s \in \{\mathsf{c}, \mathsf{p}, \mathsf{e}\}$, we have $A\{^{u'}/_{u}\} \approx_t^s B\{^{u'}/_{u}\}$.*

Proof. By induction on the derivation. □

The previous lemma indicates that the trace equivalence are preserved by replacement of free names. As for the previous proposition, the proof of Theorem 1 is taken from [15] for the classical semantics and we adapt it for the private and eavesdropping semantics.

Theorem 1. *$\approx_t^s \subsetneq \approx_m^s$ and $\approx_t^s = \approx_m^s$ on image-finite processes for $s \in \{\mathsf{c}, \mathsf{e}, \mathsf{p}\}$.*

Proof. We first prove that for all $s \in \{\mathsf{c}, \mathsf{p}, \mathsf{e}\}$, $\approx_t^s \subseteq \approx_m^s$. Since we already proved in the body of the paper that there exists two closed honest extended processes such that $A \approx_m^s B$ but $A \approx_t^s B$, we would thus obtain that $\approx_t^s \subsetneq \approx_m^s$.

Let A, B be two closed extended processes such that $A \approx_t^s B$. Let $C[\_]$ be an evaluation context s-closing for A and B, and c be a channel name. We assume w.l.o.g. that $C[\_] = \nu\tilde{n}.(D_1 \mid \nu\tilde{m}.(\_ \mid D_2))$ for some extended processes D, D' and for some sequences of names and variables $\tilde{n}$, and $\tilde{m}$. We assume w.l.o.g. that $\tilde{m} \cap (bn(A) \cup bv(A)) = \emptyset$ and $\tilde{m} \cap (bn(B) \cup bv(B)) = \emptyset$.

Let $A_2 = A\{^{\tilde{m}'}/_{\tilde{m}}\}$ and $B_2 = B\{^{\tilde{m}'}/_{\tilde{m}}\}$ where $\tilde{m}'$ is a sequence of fresh names and variables. Thanks to Lemma 10, we have that $A_2 \approx_t^s B_2$. Hence, by structural equivalence, there exists $C_2[\_] = \nu\tilde{k}.(D \mid \_)$ such that $C[A] \equiv C_2[A_2]$ and $C[B] \equiv C_2[B_2]$.

Assume now that $C[A] \Downarrow_c^s$. This means that there exist a evaluation context C_1 that does not bind c, a term M, and a plain process P, $\theta \in \{\mathsf{at}, \mathsf{ho}\}$ such that $C[A] \equiv C_2[A_2] \rightarrow_s^* C_1[\mathsf{out}^\theta(c, M).P]$. Applying Proposition 3 on A_2, B_2 and $C_2[\_]$, we know that there exist a closed extended process A_2', an evaluation context $C_2'[\_] = \nu\tilde{r}.(E \mid \_)$ s-closing for A_2' and $\mathsf{tr} \in (\mathcal{A} \setminus \{\tau\})^*$ such that $C_1[\mathsf{out}^\theta(c, M).P] \equiv C_2[A_2']$, and $A_2 \overset{\mathsf{tr}}{\Rightarrow}_s A_2'$, and for all closed extended process B_2' such that $B_2 \overset{\mathsf{tr}}{\Rightarrow}_s B_2'$ and $\phi(B_2') \sim \phi(A_2')$, we have that $C_2[B_2] \rightarrow_s^* C_2'[B_2']$. Moreover, we assume w.l.o.g. that $bn(()\mathsf{tr}) \cap fn(()B_2) = \emptyset$.

Since $C_2' = \nu\tilde{r}.(E|\_)$, we can deduce from $C_1[\mathsf{out}^\theta(c, M).P] \equiv C_2'[A_2']$ that the output $\mathsf{out}^\theta(c, M)$ comes from the process E when $\theta = \mathsf{at}$ or from A_2' when $\theta = \mathsf{ho}$. We distinguish these two cases:

- *Case* $\theta = \mathsf{at}$: Since, we have that $A_2 \approx_t^s B_2$, we know that there exists B_2' such that $B_2 \overset{\mathsf{tr}}{\Rightarrow}_s B_2'$ and $\phi(A_2') \sim \phi(B_2')$. Therefore, we have that $C_2[B_2] \rightarrow_s^* C_2'[B_2'] \equiv \nu\tilde{r}.(E \mid B_2')$. But by hypothesis, we know that the output $\mathsf{out}^\theta(c, M)$ comes from E and $c \notin \tilde{r}$. Hence we have that $C_2[B_2] \Downarrow_c^s$, and since $C[B] \equiv C_2[B_2]$, we conclude that $C[B] \Downarrow_c^s$.
- *Case* $\theta = \mathsf{ho}$: Thus, we have that $A_2' \equiv \nu\tilde{v}.(\mathsf{out}^\theta(c, M).P \mid A_3)$ with $c \notin \tilde{v}, \tilde{r}$. Thus, we have that $A_2' \xrightarrow{\nu z.out(c,z)}_s \nu\tilde{v}.(P \mid A_3 \mid \{^M/_z\})$ where z is fresh (if M is a term of channel type, the transition is different but the proof can be done in a similar way.) Let $A'' = \nu\tilde{v}.(P \mid A_3 \mid \{^M/_z\})$ and $\mathsf{tr}' = \mathsf{tr} \cdot \nu z.out(c, z)$, we have that $A_2 \overset{\mathsf{tr}'}{\Rightarrow}_s A''$. Since we have that $A_2 \approx_t^s B_2$, we have that there exists B_2' such that $B_2 \overset{\mathsf{tr}'}{\Rightarrow}_s B_2'$ and $\phi(A'') \sim \phi(B_2')$. Since internal reduction rules do not modify the frame (modulo structural equivalence), we can deduce w.l.o.g. that there exists B' such that $B_2 \overset{\mathsf{tr}}{\Rightarrow}_s B' \xrightarrow{\nu z.out(c,z)}_s B_2'$. Therefore, we have that there exists a term N, an evaluation context C_3 and a process Q such that $B' \equiv C_3[\mathsf{out}^{\mathsf{ho}}(c, N).Q]$ and c is not bind by C_3. Furthermore, we have that $\phi(A_2') \sim \phi(B')$ which means that $C_2[B_2] \rightarrow_s^* C_2'[B']$, and thus $C_2[B_2] \rightarrow_s^* C_2'[C_3[\mathsf{out}^{\mathsf{ho}}(c, N).Q]]$. Hence, we have that $C_2[B_2] \Downarrow_c^s$, and since $C[B] \equiv C_2[B_2]$, we conclude that $C[B] \Downarrow_c^s$.

This conclude the proof of $\approx_t^s \subseteq \approx_m^s$. It remains to prove that on imagine-finite processes, $\approx_t^s = \approx_m^s$ for all $s \in \{\mathsf{c}, \mathsf{e}, \mathsf{p}\}$. We first focus on $s = \mathsf{c}$.

Assume that $A \not\approx_t^{\mathsf{c}} B$. We assume w.l.o.g. that $A \not\sqsubseteq_t^s B$. In such a case, there exists a witness for the non equivalence. This means that there exists A', tr such that $bn(()\mathsf{tr}) \cap fn(()B) = \emptyset$, and for all B', $B \overset{\mathsf{tr}}{\Rightarrow}_{\mathsf{c}} B'$ implies $\phi(A') \not\sim \phi(B')$. Moreover, we assume that no name in tr is bound twice (*i.e.* $\nu a.$ cannot occur twice in tr) and bound names in tr are distinct from free names that occur in A, B, and tr.

We build an evaluation context $C_{\mathsf{c}}[\_]$ according to the trace tr and also the tests that witness the fact that static equivalence does not hold. Let $S_{\mathsf{tr}} = \{\phi(B') \mid B \overset{\mathsf{tr}}{\Rightarrow}_{\mathsf{c}} B'\}$. Since B is image-finite, we know that $S_{\mathsf{tr}}/\sim$ is finite. Let $\{\phi_1, \ldots, \phi_m\} = S/\sim$. Note that m can be equal to 0 if there is no B' such that $B \overset{\mathsf{tr}}{\Rightarrow}_{\mathsf{c}} B'$.

We know that $\{1, \ldots, m\} = T^+ \uplus T^-$ with:

- for each $i \in T^+$, there exist two terms M_i and N_i such that $v(M_i) \cup v(N_i) \subseteq \mathrm{dom}(\phi(A'))$, $(M_i =_{\mathsf{E}} N_i)\phi(A')$, and $(M_i \neq_{\mathsf{E}} N_i)\phi_i$; and
- for each $i \in T^-$, there exist two terms M_i and N_i such that $v(M_i) \cup v(N_i) \subseteq \mathrm{dom}(\phi(A'))$, $(M_i \neq_{\mathsf{E}} N_i)\phi(A')$, and $(M_i =_{\mathsf{E}} N_i)\phi_i$.

Let bad be a fresh channel name that does not occur in A and B. Let $P_1, \ldots, P_m, P_{m+1}$ be the plain processes defined as follows:

- $P_{m+1} \mathrel{\hat{=}} \mathsf{out}^{\mathsf{at}}(\mathsf{bad}, \mathsf{bad}).0$
- for $1 \leqslant i \leqslant m$, we define P_i as follows:

$$\begin{array}{ll} P_i \mathrel{\hat{=}} \mathsf{if}\ M_i = N_i\ \mathsf{then}\ P_{i+1}\ \mathsf{else}\ 0 & \text{when } i \in T^+ \\ P_i \mathrel{\hat{=}} \mathsf{if}\ M_i = N_i\ \mathsf{then}\ 0\ \mathsf{else}\ P_{i+1} & \text{when } i \in T^- \end{array}$$

Let $\{a_1, \ldots, a_k\}$ be channel names that occur free in A, B, and tr. Let $\mathcal{X}_{ch}^0 = \{x_{a_1}, \ldots, x_{a_k}\}$ be a set of variables of channel type, and $\sigma = \{x_{a_1} \mapsto a_1, \ldots, x_{a_k} \mapsto a_k\}$. Moreover, for all channel names $\{d_1, \ldots, d_m\}$ that are bound in tr, we also associate fresh variables $x_{d_1}, \ldots, x_{d_m}$.

We define $C_{\mathsf{c}}[\_]$ such that $C_{\mathsf{c}}[\_] = \nu\tilde{z}.(\mathsf{Q}_{\mathsf{c}}(\mathsf{tr}, \mathcal{X}_{ch}^0) \mid \_)$ where $\tilde{z} = \mathrm{dom}(\phi(A))$ and $\mathsf{Q}_{\mathsf{c}}(\mathsf{tr}, \mathcal{X}_{ch})$ is defined by recurrence on tr as follows:

- if $\mathsf{tr} = \epsilon$ then $\mathsf{Q}_{\mathsf{c}}(\mathsf{tr}, \mathcal{X}_{ch}) = P_1$;
- if $\mathsf{tr} = in(a, M).\mathsf{tr}'$ then $\mathsf{Q}_{\mathsf{c}}(\mathsf{tr}, \mathcal{X}_{ch}) = \mathsf{out}^{\mathsf{at}}(x_a\sigma, M).\mathsf{Q}_{\mathsf{c}}(\mathsf{tr}', \mathcal{X}_{ch})$;
- if $\mathsf{tr} = \nu z.out(a, z).\mathsf{tr}'$ and z is of base type then $\mathsf{Q}_{\mathsf{c}}(\mathsf{tr}, \mathcal{X}_{ch}) = \mathsf{in}^{\mathsf{at}}(x_a\sigma, x).\mathsf{Q}_{\mathsf{c}}(\mathsf{tr}', \mathcal{X}_{ch})$
- it $\mathsf{tr} = out(a, c).\mathsf{tr}'$ then $\mathsf{Q}_{\mathsf{c}}(\mathsf{tr}, \mathcal{X}_{ch}) = \mathsf{in}^{\mathsf{at}}(x_a\sigma, y).\mathsf{if}\ y = x_c\sigma\ \mathsf{then}\ \mathsf{Q}_{\mathsf{c}}(\mathsf{tr}', \mathcal{X}_{ch})\ \mathsf{else}\ 0$ where y is fresh variable of channel type; and
- if $\mathsf{tr} = \nu c.out(a, c)$ and c is of channel type then $\mathsf{Q}_{\mathsf{c}}(\mathsf{tr}, \mathcal{X}_{ch}) = \mathsf{in}^{\mathsf{at}}(x_a\sigma, x_c).\mathsf{if}\ x_c \in \mathcal{X}_{ch}\sigma\ \mathsf{then}\ 0\ \mathsf{else}\ \mathsf{Q}_{\mathsf{c}}(\mathsf{tr}', \mathcal{X}'_{ch})$ where $\mathcal{X}'_{ch} = \mathcal{X}_{ch} \uplus \{x_c\}$.

We use the conditional $\mathsf{if}\ u \in \{u_1, \ldots, u_k\}\ \mathsf{then}\ 0\ \mathsf{else}\ P$ as a shortcut for

$$\mathsf{if}\ u = u_1\ \mathsf{then}\ 0\ \mathsf{else}\ (\mathsf{if}\ u = u_2\ \mathsf{then}\ 0\ \mathsf{else}\ (\ldots(\mathsf{if}\ u = u_k\ \mathsf{then}\ 0\ \mathsf{else}\ P)\ldots)).$$

We can see that $C_{\mathsf{c}}[A] \Downarrow_{\mathsf{bad}}^{\mathsf{c}}$ since $A \overset{\mathsf{tr}}{\Rightarrow} A'$ and $\phi(A')$ satisfies by definition all the tests that are tested in $P_1, \ldots, P_m$. However, by construction of $C_{\mathsf{c}}[\_]$, we have that $C_{\mathsf{c}}[B] \not\Downarrow_{\mathsf{bad}}^{\mathsf{c}}$.

This conclude the proof for the case $s = \mathsf{c}$. The proof for $s = \mathsf{p}$ and e are very similar. We only need to slightly modify the context $C_{\mathsf{c}}[\_]$. In fact since the possible labels in the private semantics are the same as in the original semantics, we have $C_{\mathsf{p}}[\_] = C_{\mathsf{c}}[\_]$. However, for the eavesdropping semantics, we define $C_{\mathsf{e}}[\_]$ such that $C_{\mathsf{e}}[\_] = \nu\tilde{z}.(\mathsf{Q_e}(\mathsf{tr}, \mathcal{X}^0_{ch}) \mid \_)$ where $\tilde{z} = \mathrm{dom}(\phi(A))$ and $\mathsf{Q_e}(\mathsf{tr}, \mathcal{X}_{ch})$ is defined by recurrence on tr as follows:

- if $\mathsf{tr} = \epsilon$ then $\mathsf{Q_e}(\mathsf{tr}, \mathcal{X}_{ch}) = P_1$;
- if $\mathsf{tr} = in(a, M).\mathsf{tr}'$ then $\mathsf{Q_e}(\mathsf{tr}, \mathcal{X}_{ch}) = \mathsf{out^{at}}(x_a\sigma, M).\mathsf{Q_e}(\mathsf{tr}', \mathcal{X}_{ch})$;
- if $\mathsf{tr} = \nu z.out(a, z).\mathsf{tr}'$ and z is of base type then $\mathsf{Q_e}(\mathsf{tr}, \mathcal{X}_{ch}) = \mathsf{in^{at}}(x_a\sigma, z).\mathsf{Q_e}(\mathsf{tr}', \mathcal{X}_{ch})$
- it $\mathsf{tr} = out(a, c).\mathsf{tr}'$ then $\mathsf{Q_e}(\mathsf{tr}, \mathcal{X}_{ch}) = \mathsf{in^{at}}(x_a\sigma, y).\mathsf{if}\ y = x_c\sigma\ \mathsf{then}\ \mathsf{Q_e}(\mathsf{tr}', \mathcal{X}_{ch})\ \mathsf{else}\ 0$ where y is fresh variable of channel type; and
- if $\mathsf{tr} = \nu c.out(a, c)$ and c is of channel type then $\mathsf{Q_e}(\mathsf{tr}, \mathcal{X}_{ch}) = \mathsf{in^{at}}(x_a\sigma, x_c).\mathsf{if}\ x_c \in \mathcal{X}_{ch}\sigma\ \mathsf{then}\ 0\ \mathsf{else}\ \mathsf{Q_e}(\mathsf{tr}', \mathcal{X}'_{ch})$ where $\mathcal{X}'_{ch} = \mathcal{X}_{ch} \uplus \{x_c\}$.
- if $\mathsf{tr} = eav(a, c).\mathsf{tr}'$ with c of channel-type then $\mathsf{Q_e}(\mathsf{tr}, \mathcal{X}_{ch}) = \mathsf{eav}(x_a\sigma, y).\mathsf{if}\ y = x_c\sigma\ \mathsf{then}\ \mathsf{Q_e}(\mathsf{tr}', \mathcal{X}_{ch})\ \mathsf{else}\ 0$ where y is fresh variable of channel type;
- if $\mathsf{tr} = \nu z.eav(a, z).\mathsf{tr}'$ and z is of base type then $\mathsf{Q_e}(\mathsf{tr}, \mathcal{X}_{ch}) = \mathsf{eav}(x_a\sigma, z).\mathsf{Q_e}(\mathsf{tr}', \mathcal{X}_{ch})$
- if $\mathsf{tr} = \nu c.eav(a, c).\mathsf{tr}'$ and c is of channel type then $\mathsf{Q_e}(\mathsf{tr}, \mathcal{X}_{ch}) = \mathsf{eav}(x_a\sigma, x_c).\mathsf{if}\ x_c \in \mathcal{X}_{ch}\sigma\ \mathsf{then}\ 0\ \mathsf{else}\ \mathsf{Q_e}(\mathsf{tr}', \mathcal{X}'_{ch})$ where $\mathcal{X}'_{ch} = \mathcal{X}_{ch} \uplus \{x_c\}$. □

Appendix D. Proof of Theorem 6

In this section, we want to prove that $\approx^{\mathsf{e}}_m \subsetneq \approx^{\mathsf{p}}_m \cap \approx^{\mathsf{c}}_m$. In order to show that $\approx^{\mathsf{e}}_m \subsetneq \approx^{\mathsf{c}}_m$, we need to build a transformation of context that would allows us to go from the classic semantics to eavesdropping semantics, and vice versa.

Notice that in the definition of structural equivalence $!A \ \ |!A$ is not equivalence to $!A$ even though they have the same behavior. In fact, for reachability, may equivalence, trace equivalence, observational equivalence and labeled bissimilar, using the structural equivalence coincides with using the structural equivalence augmented with the equality $!A \ \ |!A \ \equiv !A$. As such in this section, we will consider the structural equivalence augmented with the equality $!A \ \ |!A \equiv !A$.

Definition 13. Let P be an extended attacker process. We define $\overline{P}$ inductively as follows:

- 0 when $P = 0$
- $\overline{P_1} \mid \overline{P_2}$ when $P = P_1 \mid P_2$
- P when $P = \{^u/_x\}$
- ωc when $P = \omega c$
- $\nu n.(\overline{P'}\ |!\mathsf{eav}(n, y)\ |!\mathsf{eav}(n, z))$ when $P = \nu n.P'$, n is of channel type and y, z are variables of base and channel type respectively.
- $\nu k.\overline{P'}$ when $P = \nu n.P'$, n is of base type
- $\mathsf{if}\ u = v\ \mathsf{then}\ \overline{P_1}\ \mathsf{else}\ P_2$ when $P = \mathsf{if}\ u = v\ \mathsf{then}\ P_1\ \mathsf{else}\ P_2$
- $\mathsf{eav}(c, x).\overline{P'}$ when $P = \mathsf{eav}(c, x).P'$
- $\mathsf{out^{at}}(c, u).\overline{P'}$ when $P = \mathsf{out^{at}}(c, u).P$
- $\mathsf{in^{at}}(c, x).\overline{P'}$ when $P = \mathsf{in^{at}}(c, x).P'$ and x is of base type
- $\mathsf{in^{at}}(c, x).(\overline{P'}\ |!\mathsf{eav}(x, y)\ |!\mathsf{eav}(x, z))$ when $P = \mathsf{in^{at}}(c, x).P'$, y, z are variables of base and channel type respectively.

Let $\mathcal{T}_{ch}$ be the terms of channel type, i.e. names and variables of channel type. Let $C[\_] = \nu\tilde{n}.(D \mid \_)$ be an attacker evaluation context and S a set of channel names. We define $\overline{C}_S[\_]$ as follows:

$$\nu\tilde{n}.\left(\overline{D} \mid \_ \mid \prod_{a\in\tilde{n}\cap\mathcal{T}_{ch}} !\mathsf{eav}(a, y) \mid !\mathsf{eav}(a, z) \mid \omega a\right) \mid \prod_{a\in S} !\mathsf{eav}(a, y) \mid !\mathsf{eav}(a, z) \mid \omega a$$

where y and z are variables of base and channel type respectively.

In order to facilitate the readability of the proof, for a set S of names and variables, we will denote $\mathbb{P}(S) = \prod_{a\in S\cap\mathcal{T}_{ch}} !\mathsf{eav}(a, y) \mid !\mathsf{eav}(a, z)$ and $\mathbb{P}_o(S) = \prod_{a\in S\cap\mathcal{T}_{ch}} !\mathsf{eav}(a, y) \mid !\mathsf{eav}(a, z) \mid \omega a$. Moreover, we will consider that $\mathbb{P}(S) \mid \mathbb{P}(S) \equiv \mathbb{P}(S)$.

Hence, $\overline{C}_S[\_]$ can now be written as $\nu\tilde{n}.(\overline{D} \mid \_ \mid \mathbb{P}_o(\tilde{n})) \mid \mathbb{P}_o(S)$.

Note that from the definition, we have that for all A closed honest extended process, if $C[\_] = \nu\tilde{n}.(D \mid \_)$ is c-closing for A then $\overline{C}_S[\_]$ is e-closing for A for all S.

Lemma 11. *Let A be an extended process and $\nu\tilde{n}$ a sequence of names and variables. We have $\overline{\nu\tilde{n}.A} \equiv \nu\tilde{n}.(\overline{A} \mid \mathbb{P}(\tilde{n})$.*

Proof. Direct from the definition. □

Lemma 12. *Let A be an closed honest extended process. Let $C[\_] = \nu\tilde{n}.(\nu\tilde{m}.D \mid \_)$ be an attacker evaluation context c-closing for A such that D is named-cleaned and eavesdrop-free. Let S be a set of channel names such that $fc(C[A]) \subseteq S$.*

1. *For all $C[A] \rightarrow_{\mathsf{c}} A_0$, there exist A' closed honest extended process, $C'[\_] = \nu\tilde{n}'.(\nu\tilde{m}'.D' \mid \_)$ an attacker evaluation context c-closing for A' such that D' is name-cleaned and eavesdrop-free, $C'[A'] \equiv A_0$ and $\overline{C}_S[A] \rightarrow_{\mathsf{e}} \overline{C'}_S[A']$*
2. *For all $\overline{C}_S[A] \rightarrow_{\mathsf{e}} A_0$, there exist A' closed honest extended process, $C'[\_] = \nu\tilde{n}'.(\nu\tilde{m}'.D' \mid \_)$ an attacker evaluation context c-closing for A' such that D' is name-cleaned and eavesdrop-free, $\overline{C'}_S[A'] \equiv A_0$ and $C[A] \rightarrow_{\mathsf{c}} C'[A']$*

Proof. We first start by proving the first property. Notice that by structural equivalence, we can always assume that the bound names and variables in $C[A]$ are only bound once and are distinct from the free names in S. Indeed, for all $C''[\_]$, A'', if $C[A] \equiv C''[A'']$ only by renaming of bound names and variables then we obtain that $\overline{C}_S[A] \equiv \overline{C''}_S[A'']$.

We do a case analysis on the internal rule applied.

Case 1.a, rule THEN *on D, i.e. $D =$* if $u = v$ then D_1 else $D_2 \mid D_3$ *and $A_0 \equiv \nu\tilde{n}.(D_2 \mid D_3 \mid A)$*: In such a case we have $\overline{D} \rightarrow_{\mathsf{e}} \overline{D_1} \mid \overline{D_3}$ and so $\nu\tilde{m}.(\overline{D} \mid \mathbb{P}(\tilde{m})) \rightarrow_{\mathsf{e}} \nu\tilde{m}.(\overline{D_1} \mid \overline{D_3} \mid \mathbb{P}(\tilde{m}))$. By Lemma 11, we obtain that $\overline{\nu\tilde{m}.D} \rightarrow \overline{\nu\tilde{m}.(D_1 \mid D_3)}$. Let us denote $C'[\_] = \nu\tilde{n}.(\nu\tilde{m}.(D_1 \mid D_3) \mid \_)$ and $A' = A$. Since $\overline{C'}_S[\_] = \nu\tilde{n}.(\nu\tilde{m}.(D_1 \mid D_3) \mid \_ \mid \mathbb{P}_o(\tilde{n})) \mid \mathbb{P}_o(S)$, we obtain that $A_0 \equiv C'[A']$ and $\overline{C}_S[A] \rightarrow_{\mathsf{e}} \overline{C'}_S[A']$.

Case 1.b, rule ELSE *on D*: Similar to Case 1.a.

Case 2.a, rule THEN *on A, i.e. $A \equiv \nu\tilde{r}.($*if $u = v$ then P_1 else $P_2 \mid P_3)$ *and $A_0 \equiv C[\nu\tilde{r}.(P_1 \mid P_3)]$*: In such a case, let us denote $C'[\_] = C[\_]$ and $A' = \nu\tilde{r}.(P_1 \mid P_3)$. Therefore, $C'[A'] = C[A'] \equiv A_0$. Note that $A \rightarrow_{\mathsf{e}} A'$. Hence $C[A] \rightarrow_{\mathsf{e}} C[A']$ and $\overline{C}_S[A] \rightarrow_{\mathsf{e}} \overline{C}_S[A']$. Thus the result holds.

Case 2.b, rule ELSE *on A*: Similar to Case 2.a.

Case 3, rule COMM *on A, i.e. $A \equiv \nu\tilde{r}.(\mathsf{out}^{\mathsf{ho}}(c,u).P_1 \mid \mathsf{in}^{\mathsf{ho}}(c,x).P_2 \mid P_3)$ and $A_0 \equiv C[\nu\tilde{r}.(P_1 \mid P_2\{^u/_x\} \mid P_3)]$*: Note that even though $A \rightarrow_{\mathsf{c}} \nu\tilde{r}.(P_1 \mid P_2\{^u/_x\} \mid P_3)$, we don't necessarily have that $A \rightarrow_{\mathsf{e}} \nu\tilde{r}.(P_1 \mid P_2\{^u/_x\} \mid P_3)$. We have to do a case analysis on u and c:

- Case 3.a, $c \in \tilde{r}$: In such a case, we know from A being an honest processes that $c \notin oc(P_3)$. Thus we can apply rule C-PRIV to obtain that $A \rightarrow_{\mathsf{e}} \nu\tilde{r}.(P_1 \mid P_2\{^u/_x\} \mid P_3)$. Hence, by denoting $C'[\_] = C[\_]$ and $A' = \nu\tilde{r}.(P_1 \mid P_2\{^u/_x\} \mid P_3)$, we obtain that $C'[A'] = C[A'] \equiv A_0$, $A \rightarrow_{\mathsf{e}} A'$ and so $C[A] \rightarrow_{\mathsf{e}} C[A']$ and $\overline{C}_S[A] \rightarrow_{\mathsf{e}} \overline{C}_S[A']$. Therefore, the result holds.
- Case 3.b, $c \notin \tilde{r}$ and u of base type: In such a case, $\mathsf{out}^{\mathsf{ho}}(c,u).P_1 \mid \mathsf{in}^{\mathsf{ho}}(c,x).P_2 \mid P_3 \mid \mathsf{eav}(c,y) \rightarrow_{\mathsf{e}} P_1 \mid P_2\{^u/_x\} \mid P_3$ by the rule C-EAV. Let us denote $A' = \nu\tilde{r}.(P_1 \mid P_2\{^u/_x\} \mid P_3)$. Since $c \notin \tilde{r}$, we obtain that $A \mid \mathsf{eav}(c,y) \rightarrow_{\mathsf{e}} A'$ and so $A\ |!\mathsf{eav}(c,y) \rightarrow_{\mathsf{e}} A'\ |!\mathsf{eav}(c,y)$. By noticing that c is either in $\tilde{n}$ or in $fc(C[A])$ and so in S, the structural equivalence gives us that $\overline{C}_S[A] \rightarrow_{\mathsf{e}} \overline{C}_S[A']$. Hence the result holds with $C'[\_] = C[\_]$.
- Case 3.c, $c \notin \tilde{r}$ and u of channel type: This case is very similar to Case 3.b. Indeed, $\mathsf{out}^{\mathsf{ho}}(c,u).P_1 \mid \mathsf{in}^{\mathsf{ho}}(c,x).P_2 \mid P_3 \mid \mathsf{eav}(c,z) \rightarrow_{\mathsf{e}} P_1 \mid P_2\{^u/_x\} \mid P_3 \mid \omega c$ by the rule C-OEAV. Let us denote $A' = \nu\tilde{r}.(P_1 \mid P_2\{^u/_x\} \mid P_3)$. Since $c \notin \tilde{r}$, we obtain that $A \mid \mathsf{eav}(c,z) \rightarrow_{\mathsf{e}} A' \mid \omega c$ and so $A\ |!\mathsf{eav}(c,z) \mid \omega c \rightarrow_{\mathsf{e}} A'\ |!\mathsf{eav}(c,z) \mid \omega c$. By noticing that c is either in $\tilde{n}$ or in $fc(C[A])$ and so in S, the structural equivalence gives us that $\overline{C}_S[A] \rightarrow_{\mathsf{e}} \overline{C}_S[A']$. Hence the result holds with $C'[\_] = C[\_]$.

Case 4, rule COMM *on D, i.e. $D = \mathsf{out}^{\mathsf{at}}(c,u).D_1 \mid \mathsf{in}^{\mathsf{at}}(c,x).D_2 \mid D_3$ and $A_0 \equiv \nu\tilde{n}.(\nu\tilde{m}.(D_1 \mid D_2\{^u/_x\} \mid D_3) \mid A)$*: Let us do a case analysis on u:

- Case 4.a, u is of base type: In such a case, we have $\mathsf{out}^{\mathsf{at}}(c,u).\overline{D_1} \mid \mathsf{in}^{\mathsf{at}}(c,x).\overline{D_2} \mid \overline{D_3} \rightarrow_{\mathsf{e}} \overline{D_1} \mid \overline{D_2\{^u/_x\}} \mid \overline{D_3}$ by the rule C-ENV. Hence, $\nu\tilde{m}.(\mathsf{out}^{\mathsf{at}}(c,u).\overline{D_1} \mid \mathsf{in}^{\mathsf{at}}(c,x).\overline{D_2} \mid \overline{D_3} \mid \mathbb{P}(\tilde{m})) \rightarrow_{\mathsf{e}} \nu\tilde{m}.(\overline{D_1} \mid \overline{D_2\{^u/_x\}} \mid \overline{D_3} \mid \mathbb{P}(\tilde{m}))$. Let us denote $D' = (D_1 \mid D_2\{^u/_x\} \mid D_3)$. By Lemma 11, we obtain that $\overline{\nu\tilde{m}.D} \rightarrow_{\mathsf{e}} \overline{\nu\tilde{m}.D'}$. Hence, we deduce that $\nu\tilde{n}.(\overline{\nu\tilde{m}.D} \mid A \mid \mathbb{P}_o(\tilde{n})) \mid \mathbb{P}_o(S) \rightarrow_{\mathsf{e}} \nu\tilde{n}.(\overline{\nu\tilde{m}.D'} \mid A \mid \mathbb{P}_o(\tilde{n})) \mid \mathbb{P}_o(S)$. Let us denote $C'[\_] = \nu\tilde{n}.(\nu\tilde{m}.D' \mid \_)$ and $A' = A$. We have $A_0 \equiv C'[A']$ and $\overline{C}_S[A] \rightarrow_{\mathsf{e}} \overline{C'}_S[A']$. Hence the result holds.
- Case 4.b, u is of channel type and $u \notin \tilde{m} \cup \tilde{n}$: In such a case, $u \in fv(C[A]) \subseteq S$ and we have $\mathsf{out}^{\mathsf{at}}(c,u).\overline{D_1} \mid \mathsf{in}^{\mathsf{at}}(c,x).(\overline{D_2} \mid \mathbb{P}(x)) \mid \overline{D_3} \rightarrow_{\mathsf{e}} \overline{D_1} \mid \overline{D_2\{^u/_x\}} \mid \overline{D_3} \mid \mathbb{P}(u) \mid \omega u$ by the rule C-OPEN. Since $u \notin \tilde{m}$, we obtain that $\nu\tilde{m}.(\mathsf{out}^{\mathsf{at}}(c,u).\overline{D_1} \mid \mathsf{in}^{\mathsf{at}}(c,x).(\overline{D_2} \mid \mathbb{P}(x)) \mid \overline{D_3} \mid \mathbb{P}(\tilde{m})) \rightarrow_{\mathsf{e}} \nu\tilde{m}.(\overline{D_1} \mid \overline{D_2\{^u/_x\}} \mid \overline{D_3} \mid \mathbb{P}(\tilde{m})) \mid \mathbb{P}_o(u)$. Let us denote $D' = (D_1 \mid D_2\{^u/_x\} \mid D_3)$. By Lemma 11, we obtain that $\overline{\nu\tilde{m}.D} \rightarrow_{\mathsf{e}} \overline{\nu\tilde{m}.D'} \mid \mathbb{P}_o(u)$. Moreover, since $u \notin \tilde{n}$ then $\nu\tilde{n}.(\overline{\nu\tilde{m}.D} \mid A \mid \mathbb{P}_o(\tilde{n})) \rightarrow_{\mathsf{e}} \tilde{n}.(\overline{\nu\tilde{m}.D'} \mid A \mid \mathbb{P}_o(\tilde{n})) \mid \mathbb{P}_o(u)$. Lastly, since $u \in S$ and $\mathbb{P}_o(u) \mid \mathbb{P}_o(u) \equiv \mathbb{P}_o(u)$, we obtain that $\nu\tilde{n}.(\overline{\nu\tilde{m}.D} \mid A \mid \mathbb{P}_o(\tilde{n})) \mid \mathbb{P}_o(S) \rightarrow_{\mathsf{e}} \tilde{n}.(\overline{\nu\tilde{m}.D'} \mid A \mid \mathbb{P}_o(\tilde{n})) \mid \mathbb{P}_o(S)$. Therefore, the result holds with $A' = A$ and $C'[\_] = \nu\tilde{n}.(\nu\tilde{m}.D' \mid \_)$.
- Case 4.c, u is of channel type and $u \in \tilde{n}$: This case is similar to Case 4.b. Since $u \notin \tilde{m}$, we can apply the same reasoning and obtain $\overline{\nu\tilde{m}.D} \rightarrow_{\mathsf{e}} \overline{\nu\tilde{m}.D'} \mid \mathbb{P}_o(u)$ where $D' = (D_1 \mid D_2\{^u/_x\} \mid D_3)$. Since $u \in \tilde{n}$ and $\mathbb{P}_o(u) \mid \mathbb{P}_o(u) \equiv \mathbb{P}_o(u)$, we deduce that $\nu\tilde{n}.(\overline{\nu\tilde{m}.D} \mid A \mid \mathbb{P}_o(\tilde{n})) \rightarrow_{\mathsf{e}} \nu\tilde{n}.(\overline{\nu\tilde{m}.D'} \mid A \mid \mathbb{P}_o(\tilde{n}))$. Therefore, we obtain that $\nu\tilde{n}.(\overline{\nu\tilde{m}.D} \mid A \mid \mathbb{P}_o(\tilde{n})) \mid \mathbb{P}_o(S) \rightarrow_{\mathsf{e}} \nu\tilde{n}.(\overline{\nu\tilde{m}.D'} \mid A \mid \mathbb{P}_o(\tilde{n})) \mid \mathbb{P}_o(S)$ and so the result holds with $A' = A$ and $C'[\_] = \nu\tilde{n}.(\nu\tilde{m}.D' \mid \_)$.
- Case 4.d, u is of channel type and $u \in \tilde{m}$: First of all, note that since $u \in \tilde{m}$, $\nu\tilde{m}.D \equiv \nu u.\nu\tilde{m}'.D$ for some $\tilde{m}'$ such that $u \notin \tilde{m}'$. Note that since u is bound, $u \notin fv(A) \cup fn(A)$. Hence, by applying the same reasoning as in Case 4.b, we obtain that $\overline{\nu\tilde{m}'.D} \rightarrow_{\mathsf{e}} \overline{\nu\tilde{m}'.D'} \mid \mathbb{P}_o(u)$ where $D' = (D_1 \mid D_2\{^u/_x\} \mid D_3)$. Since $\mathbb{P}(u) \mid \mathbb{P}_o(u) \equiv \mathbb{P}(u) \mid \omega u \mid \mathbb{P}(u) \equiv \mathbb{P}_o(u)$, we deduce that $\nu u.(\overline{\nu\tilde{m}'.D} \mid \mathbb{P}(u)) \rightarrow_{\mathsf{e}} \nu u.(\overline{\nu\tilde{m}'.D'} \mid \mathbb{P}_o(u))$. First, notice that $\nu u.(\overline{\nu\tilde{m}'.D} \mid \mathbb{P}(u)) = \overline{\nu u.\nu\tilde{m}'.D} = \overline{\nu\tilde{m}.D}$ by

Lemma 11. Second, since u does not appear in A, we deduce that $\nu\tilde{n}.(\overline{\nu\tilde{m}.D} \mid A \mid \mathbb{P}_o(\tilde{n})) \rightarrow_{\mathsf{e}} \nu\tilde{n}.(\nu u.(\overline{\nu\tilde{m}'.D'} \mid \mathbb{P}_o(u)) \mid A \mid \mathbb{P}_o(\tilde{n}) \equiv \nu\tilde{n}.\nu u.(\overline{\nu\tilde{m}'.D'} \mid A \mid \mathbb{P}_o(\tilde{n} \cup \{u\}))$. Hence, if we denote $\tilde{n}' = \nu\tilde{n}.\nu\tilde{u}$ then $\nu\tilde{n}.(\overline{\nu\tilde{m}.D} \mid A \mid \mathbb{P}_o(\tilde{n})) \rightarrow_{\mathsf{e}} \nu\tilde{n}'.(\overline{\nu\tilde{m}'.D'} \mid A \mid \mathbb{P}_o(\tilde{n}'))$. Therefore, by denoting $C'[\_] = \nu\tilde{n}'.(\nu\tilde{m}'.D' \mid_{)}$ and $A' = A$, we deduce $\overline{C}_S[A] \rightarrow_{\mathsf{e}} \overline{C'}_S[A']$. Thus the result holds.

Case 5, rule Comm *between A (input) and D (output), i.e.* $D = \mathsf{out}^{\mathsf{at}}(c, u).D_1 \mid D_2$, $A \equiv \nu\tilde{r}.(\mathsf{in}^{\mathsf{ho}}(c, x).P_1 \mid P_2)$ *and* $A_0 \equiv \nu\tilde{n}.\nu\tilde{m}.\nu\tilde{r}.(D_1 \mid D_2 \mid P_1\{^{u}/_{x}\} \mid P_2)$: Note that $c \notin \tilde{m} \cup \tilde{r}$. Let us do a case analysis on u:

- Case 5.a, u is of base type: In such a case, let us split $\tilde{r}$ and $\tilde{m}$ in $\tilde{r}_b.\tilde{r}_c$ and $\tilde{m}_b.\tilde{m}_c$ respectively, such that $\tilde{r}_c$ and $\tilde{m}_c$ are of channel type, and $\tilde{r}_b$ and $\tilde{m}_b$ are of base type. Since u is of base type, we deduce that $A_0 \equiv \nu\tilde{n}.\nu\tilde{m}_b.\nu\tilde{r}_b.(\nu\tilde{m}_c.(D_1 \mid D_2) \mid \nu\tilde{r}_c.(P_1\{^{u}/_{x}\} \mid P_2))$. Note that $\mathsf{out}^{\mathsf{at}}(c, u).\overline{D_1} \mid \overline{D_2} \mid \mathsf{in}^{\mathsf{ho}}(c, x).P_1 \mid P_2 \rightarrow_{\mathsf{e}} \overline{D_1} \mid \overline{D_2} \mid P_1\{^{u}/_{x}\} \mid P_2$ by the rule C-Env. Hence $\mathsf{out}^{\mathsf{at}}(c, u).\overline{D_1} \mid \overline{D_2} \mid \mathsf{in}^{\mathsf{ho}}(c, x).P_1 \mid P_2 \mid \mathbb{P}(\tilde{m}) \rightarrow_{\mathsf{e}} \overline{D_1} \mid \overline{D_2} \mid P_1\{^{u}/_{x}\} \mid P_2 \mid \mathbb{P}(\tilde{m})$. Therefore, $\nu\tilde{m}.\nu\tilde{r}.(\mathsf{out}^{\mathsf{at}}(c, u).\overline{D_1} \mid \overline{D_2} \mid \mathsf{in}^{\mathsf{ho}}(c, x).P_1 \mid P_2 \mid \mathbb{P}(\tilde{m})) \rightarrow_{\mathsf{e}} \nu\tilde{m}.\nu\tilde{r}.(\overline{D_1} \mid \overline{D_2} \mid P_1\{^{u}/_{x}\} \mid P_2 \mid \mathbb{P}(\tilde{m}))$. But $\nu\tilde{m}.\nu\tilde{r}.(\mathsf{out}^{\mathsf{at}}(c, u).\overline{D_1} \mid \overline{D_2} \mid \mathsf{in}^{\mathsf{ho}}(c, x).P_1 \mid P_2 \mid \mathbb{P}(\tilde{m})) \equiv \overline{\nu\tilde{m}.D} \mid A$ thanks to Lemma 11 and since we assume that bound names and variables are bound once and distinct from free names and variables. Moreover, $\nu\tilde{m}.\nu\tilde{r}.(\overline{D_1} \mid \overline{D_2} \mid P_1\{^{u}/_{x}\} \mid P_2 \mid \mathbb{P}(\tilde{m})) \equiv \nu\tilde{m}_b.\nu\tilde{r}_b.(\nu\tilde{m}_c.(\overline{D_1} \mid \overline{D_2} \mid \mathbb{P}(\tilde{m}_c)) \mid \nu\tilde{r}_c.(P_1\{^{u}/_{x}\} \mid P_2))$. Therefore, let us denote $\tilde{n}' = \tilde{n}.\tilde{m}_b.\tilde{r}_b$, $D' = D_1 \mid D_2$ and $A' = \nu\tilde{r}_c.(P_1\{^{u}/_{x}\} \mid P_2)$. Notice that $\tilde{m}_b$ and $\tilde{r}_b$ being of base type implies that $\mathbb{P}_o(\tilde{n}) = \mathbb{P}_o(\tilde{n}')$. Hence $\nu\tilde{n}.(\overline{\nu\tilde{m}.D} \mid A \mid \mathbb{P}_o(\tilde{n})) \mid \mathbb{P}_o(S) \rightarrow_{\mathsf{e}} \nu\tilde{n}'.(\overline{\nu\tilde{m}_c.D'} \mid A' \mid \mathbb{P}_o(\tilde{n}')) \mid \mathbb{P}_o(S)$. Hence, the result holds with $C'[\_] = \nu\tilde{n}'.(\nu\tilde{m}_c.D' \mid \_)$.
- Case 5.b, u is of channel type and $u \notin \tilde{m} \cup \tilde{n}$: Notice that in such a case $A_0 \equiv \nu\tilde{n}.(\nu\tilde{m}.(D_1 \mid D_2) \mid \nu\tilde{r}.(P_1\{^{u}/_{x}\} \mid P_2))$. The rest of the proof follows a similar reasoning as in Case 4.b and the result will hold with $C'[\_] = \nu\tilde{n}.(\nu\tilde{m}.D' \mid \_)$, $D' = D_1 \mid D_2$ and $A' = \nu\tilde{r}.(P_1\{^{u}/_{x}\} \mid P_2)$.
- Case 5.c, u is of channel type and $u \in \tilde{n}$: Notice that in such a case $A_0 \equiv \nu\tilde{n}.(\nu\tilde{m}.(D_1 \mid D_2) \mid \nu\tilde{r}.(P_1\{^{u}/_{x}\} \mid P_2))$. The rest of the proof follows a similar reasoning as in Case 4.c and the result will hold with $C'[\_] = \nu\tilde{n}.(\nu\tilde{m}.D' \mid \_)$, $D' = D_1 \mid D_2$ and $A' = \nu\tilde{r}.(P_1\{^{u}/_{x}\} \mid P_2)$.
- Case 5.d, u is of channel type and $u \in \tilde{m}$: Note that since $u \in \tilde{m}$, $\nu\tilde{m}.D \equiv \nu u.\nu\tilde{m}'.D$ for some $\tilde{m}'$ such that $u \notin \tilde{m}'$. Hence, $A_0 \equiv \nu\tilde{n}.\nu u.(\nu\tilde{m}'.(D_1 \mid D_2) \mid \nu\tilde{r}.(P_1\{^{u}/_{x}\} \mid P_2))$. The rest of the proof follows a similar reasoning as in Case 4.d and the result will hold with $C'[\_] = \nu\tilde{n}'.(\nu\tilde{m}'.D' \mid \_)$, $D' = D_1 \mid D_2$, $\tilde{n}' = \tilde{n}.u$ and $A' = \nu\tilde{r}.(P_1\{^{u}/_{x}\} \mid P_2)$.

Case 6, rule Comm *between A (output) and D (input), i.e.* $D = \mathsf{in}^{\mathsf{at}}(c, x).D_1 \mid D_2$, $A \equiv \nu\tilde{r}.(\mathsf{out}^{\mathsf{ho}}(c, u).P_1 \mid P_2)$ *and* $A_0 \equiv \nu\tilde{n}.\nu\tilde{m}.\nu\tilde{r}.(D_1\{^{u}/_{x}\} \mid D_2 \mid P_1 \mid P_2)$: Note that $c \notin \tilde{m} \cup \tilde{r}$. Let us do a case analysis on u:

- Case 6.a, u is of base type: In such a case, let us split $\tilde{r}$ and $\tilde{m}$ in $\tilde{r}_b.\tilde{r}_c$ and $\tilde{m}_b.\tilde{m}_c$ respectively, such that $\tilde{r}_c$ and $\tilde{m}_c$ are of channel type, and $\tilde{r}_b$ and $\tilde{m}_b$ are of base type. The rest of the proof follows a similar reasoning as in Case 5.a and the result holds with $C'[\_] = \nu\tilde{n}'.(\nu\tilde{m}_c.D' \mid \_)$, $\tilde{n}' = \tilde{n}.\tilde{m}_b.\tilde{r}_b$, $D' = D_1\{^{u}/_{x}\} \mid D_2$ and $A' = \nu\tilde{r}_c.(P_1 \mid P_2)$.
- Case 6.b, u is of channel type and $u \notin \tilde{m} \cup \tilde{n}$: Notice that in such a case $A_0 \equiv \nu\tilde{n}.(\nu\tilde{m}.(D_1\{^{u}/_{x}\} \mid D_2) \mid \nu\tilde{r}.(P_1 \mid P_2))$. The rest of the proof follows a similar reasoning as in Case 4.b and the result will hold with $C'[\_] = \nu\tilde{n}.(\nu\tilde{m}.D' \mid \_)$, $D' = D_1\{^{u}/_{x}\} \mid D_2$ and $A' = \nu\tilde{r}.(P_1 \mid P_2)$.
- Case 6.c, u is of channel type and $u \in \tilde{n}$: Notice that in such a case $A_0 \equiv \nu\tilde{n}.(\nu\tilde{m}.(D_1 \mid D_2) \mid \nu\tilde{r}.(P_1\{^{u}/_{x}\} \mid P_2))$. The rest of the proof follows a similar reasoning as in Case 4.c and the result will hold with $C'[\_] = \nu\tilde{n}.(\nu\tilde{m}.D' \mid \_)$, $D' = D_1\{^{u}/_{x}\} \mid D_2$ and $A' = \nu\tilde{r}.(P_1 \mid P_2)$.

- Case 6.d, u is of channel type and $u \in \tilde{m}$: Note that since $u \in \tilde{m}$, $\nu\tilde{m}.D \equiv \nu u.\nu\tilde{m}'.D$ for some $\tilde{m}'$ such that $u \notin \tilde{m}'$. Hence, $A_0 \equiv \nu\tilde{n}.\nu u.(\nu\tilde{m}'.(D_1 \mid D_2) \mid \nu\tilde{r}.(P_1\{^u/_x\} \mid P_2))$. The rest of the proof follows a similar reasoning as in Case 4.d and the result will hold with $C'[\_] = \nu\tilde{n}'.(\nu\tilde{m}'.D' \mid \_)$, $D' = D_1\{^u/_x\} \mid D_2$, $\tilde{n'} = \tilde{n}.u$ and $A' = \nu\tilde{r}.(P_1 \mid P_2)$.

This conclude the proof of the first property. The second property is in fact easy to prove: All rules in the eavesdropping semantics other than Then and Else will be mapped by the rule Comm in the classical semantics. One can notice that since we know that A and C do not contain eavesdrop processes and since the transformation from A to $\overline{A}$ and $C[\_]$ to $\overline{C}_S[\_]$ only adds processes of the form $\mathsf{eav}(c, y).0$, the communication rules all becomes instances of the rule Comm. For instance, an application of rule C-Eav would result into the following

$$\mathsf{out}^{\mathsf{ho}}(c, u).P \mid \mathsf{in}^{\mathsf{ho}}(c, x).Q \mid \mathsf{eav}(c, y).0 \xrightarrow{\tau}_{\mathsf{e}} P \mid Q\{^u/_x\}$$

which is typically the rule Comm when we remove the transformation and so the process $\mathsf{eav}(c, y).0$. Lastly, since any instance of ωd has no impact on the classical semantics, every rules thus corresponds to the rule Comm once the transformation removed. □

Corollary 3. *Let A be an closed honest extended process. Let $C[\_] = \nu\tilde{n}.(\nu\tilde{m}.D \mid \_)$ be an attacker evaluation context c-closing for A such that D is named-cleaned and eavesdrop-free. Let S be a set of channel names such that $fc(C[A]) \subseteq S$. For all channel c, $C[A] \Downarrow^{\mathsf{c}}_c$ iff $\overline{C}_S[A] \Downarrow^{\mathsf{e}}_c$.*

Theorem 6. $\approx^{\mathsf{e}}_m \subsetneq \approx^{\mathsf{p}}_m \cap \approx^{\mathsf{c}}_m$.

Proof. Consider two closed honest extended process A and B. We assume $A \approx^{\mathsf{e}}_m B$. We first show that $A \approx^{\mathsf{c}}_m B$.

Let $C[\_]$ be an attacker evaluation context c-closing for A and B. Notice that in the classical semantics, a process $\mathsf{eav}(c, x).P$ as the same behaviour as the process 0. Hence, there exists $C^1[\_]$ an attacker evaluation context eavesdrop-free and c-closing for A and B such that for all c, $C[A] \Downarrow^{\mathsf{c}}_c \Leftrightarrow C^1[A] \Downarrow^{\mathsf{c}}_c$ and $C[B] \Downarrow^{\mathsf{c}}_c \Leftrightarrow C^1[B] \Downarrow^{\mathsf{c}}_c$ (1). Moreover, relying on the structural equivalence, we deduce that there exists $C^2 = \nu\tilde{n}.(\nu\tilde{m}.D \mid \nu\tilde{r}.(\_ \mid E))$ attacker evaluation context eavesdrop-free and c-closing for A and B such that D is named-cleaned, $C^1[A] \equiv C^2[A]$ and $C^1[B] \equiv C^2[B]$. Lastly, by renaming $\tilde{r}$ through the structural equivalence, we deduce that there exist A', B' two closed honest extended process and $C^3[\_] = \nu\tilde{n}'.(\nu\tilde{m}'.(D' \mid \_))$ attacker evaluation context eavesdrop-free and c-closing for A and B such that D is named-cleaned, $C^2[A] \equiv C^3[A']$ and $C^2[B] \equiv C^3[B']$. Therefore, we have $C^1[A] \equiv C^3[A']$ and $C^1[B] \equiv C^3[B']$. Lastly, let us denote $S = fc(C^3[A']) \cup fc(C^3[B'])$, relying on Lemma 11 and Definition 13, one can note that there exists C^4 attacker evaluation context e-closing for A and B such that $\overline{C^3}_S[A'] \equiv C^4[A]$ and $\overline{C^3}_S[B'] \equiv C^4[B]$.

We can conclude the proof as follows: Let $S = fc(C[A]) \cup fc(C[B])$. For all channel c,

$$\begin{array}{lll}
 & C[A] \Downarrow^{\mathsf{c}}_c & \\
\text{iff} & C^1[A] \Downarrow^{\mathsf{c}}_c & \text{by (1)} \\
\text{iff} & C^3[A'] \Downarrow^{\mathsf{c}}_c & \text{since } C^1[A] \equiv C^3[A'] \\
\text{iff} & \overline{C^3}_S[A'] \Downarrow^{\mathsf{e}}_c & \text{by Corollary 3} \\
\text{iff} & C^4[A] \Downarrow^{\mathsf{e}}_c & \text{since } \overline{C^3}_S[A'] \equiv C^4[A] \\
\text{iff} & C^4[B] \Downarrow^{\mathsf{e}}_c & \text{since } A \approx^{\mathsf{e}}_m B \\
\text{iff} & \overline{C^3}_S[B'] \Downarrow^{\mathsf{e}}_c & \text{since } \overline{C^3}_S[B'] \equiv C^4[B] \\
\text{iff} & C^3[B'] \Downarrow^{\mathsf{c}}_c & \text{by Corollary 3} \\
\text{iff} & C^1[B] \Downarrow^{\mathsf{c}}_c & \text{since } C^1[B] \equiv C^3[B'] \\
\text{iff} & C[B] \Downarrow^{\mathsf{c}}_c & \text{by (1)}
\end{array}$$

Let us now prove that $A \approx^{\mathsf{p}}_m B$. Let $C[\_]$ be an attacker evaluation context p-closing for A and B. As for the classical semantics, notice that in the private semantics, a process $\mathsf{eav}(c, x).P$ as the same behaviour as the process 0. Hence, there exists $C^1[\_]$ an attacker evaluation context eavesdrop-free and p-closing for A and B such that for all c, $C[A] \Downarrow^{\mathsf{p}}_c \Leftrightarrow C^1[A] \Downarrow^{\mathsf{p}}_c$ and $C[B] \Downarrow^{\mathsf{p}}_c \Leftrightarrow C^1[B] \Downarrow^{\mathsf{p}}_c$. Moreover, notice that $\rightarrow_{\mathsf{p}} \subseteq \rightarrow_{\mathsf{e}}$. Hence, for all c, $C^1[A] \Downarrow^{\mathsf{p}}_c$ implies $C^1[A] \Downarrow^{\mathsf{e}}_c$ and $C^1[B] \Downarrow^{\mathsf{p}}_c$ implies $C^1[B] \Downarrow^{\mathsf{e}}_c$. Furthermore, since $C^1[\_]$ is eavesdrop-free and A, B are both honest, we deduce that rules C-EAV and C-OEAV can never be applied in a derivation of $C^1[A]$ or $C^1[B]$. Hence, we obtain that for all c, $C^1[A] \Downarrow^{\mathsf{p}}_c \Leftrightarrow C^1[A] \Downarrow^{\mathsf{e}}_c$ and $C^1[B] \Downarrow^{\mathsf{p}}_c \Leftrightarrow C^1[B] \Downarrow^{\mathsf{e}}_c$. Lastly, $A \approx^{\mathsf{e}}_m B$ implies that for all channel c, $C^1[A] \Downarrow^{\mathsf{e}}_c \Leftrightarrow C^1[B] \Downarrow^{\mathsf{e}}_c$. We can conclude the proof by combining all these statements as follows: for all channel c,

$$C[A] \Downarrow^{\mathsf{p}}_c \Leftrightarrow C^1[A] \Downarrow^{\mathsf{p}}_c \Leftrightarrow C^1[A] \Downarrow^{\mathsf{e}}_c \Leftrightarrow C^1[B] \Downarrow^{\mathsf{e}}_c \Leftrightarrow C^1[B] \Downarrow^{\mathsf{p}}_c \Leftrightarrow C[B] \Downarrow^{\mathsf{p}}_c$$

We have concluded the proof of $\approx^{\mathsf{e}}_m \subseteq \approx^{\mathsf{p}}_m \cap \approx^{\mathsf{c}}_m$. Therefore, it remains to show that this inclusion is not strict. In Fig. 7, we have provided two processes A and B such that $A \approx^{\mathsf{c}}_\ell B$, $A \approx^{\mathsf{p}}_\ell B$ but $A \not\approx^{\mathsf{e}}_t B$. Notice that these processes do not contain replication and so are imagine-finite. Thus, by Theorem 1, $A \not\approx^{\mathsf{e}}_t B$ implies $A \not\approx^{\mathsf{e}}_m B$. Moreover, by Proposition 3, $A \approx^{\mathsf{c}}_\ell B$ and $A \approx^{\mathsf{p}}_\ell B$ implies $A \approx^{\mathsf{c}}_t B$, $A \approx^{\mathsf{p}}_t B$. Once again by Theorem 1, we deduce that $A \approx^{\mathsf{c}}_m B$, $A \approx^{\mathsf{p}}_m B$. Hence, we conclude that $\approx^{\mathsf{e}}_m \subsetneq \approx^{\mathsf{p}}_m \cap \approx^{\mathsf{c}}_m$. □

Appendix E. Proof of Theorem 2

Theorem 2. *For all ground, closed honest extended processes A, for all channels d, we have that $A \downdownarrows^{\mathsf{p}}_d$ iff $A \downdownarrows^{\mathsf{c}}_d$ iff $A \downdownarrows^{\mathsf{e}}_d$.*

Proof. We will prove that the following three implications: (1) $A \downdownarrows^{\mathsf{c}}_d \Rightarrow A \downdownarrows^{\mathsf{p}}_d$, (2) $A \downdownarrows^{\mathsf{p}}_d \Rightarrow A \downdownarrows^{\mathsf{e}}_d$ and (3) $A \downdownarrows^{\mathsf{e}}_d \Rightarrow A \downdownarrows^{\mathsf{c}}_d$.

Given a trace tr, let us denote $S(\mathsf{tr}) = \{c \mid \mathsf{tr}_1 out(c, t)\mathsf{tr}_2 = \mathsf{tr}$ and tr_1 does not bind $c\}$.

Implication 1, $A \downdownarrows^{\mathsf{c}}_d \Rightarrow A \downdownarrows^{\mathsf{p}}_d$: Since A is honest, the only rules that differs are the rules COMM and C-PRIV. Furthermore, since A is honest we also know that $c \notin oc(A)$.

We show that for all $A \overset{\mathsf{tr}}{\Rightarrow}_{\mathsf{c}} A'$, there exist $\nu\tilde{n}.A'' \equiv A'$, tr' and a frame ϕ such that $S(\mathsf{tr}) \subseteq S(\mathsf{tr}')$ and $A \overset{\mathsf{tr}'}{\Rightarrow}_{\mathsf{p}} \nu\tilde{n}.(A'' \mid \phi)$ such that. We prove this result by induction on the length of the derivation $A \xrightarrow{\ell_1 \ldots \ell_m}_{\mathsf{c}} A'$ with tr being $\ell_1 \ldots \ell_m$ without the τ actions.

Base case $m = 0$: Hence $\mathsf{tr} = \varepsilon$ and so the result directly holds with $\phi = 0$.

Inductive step $m > 0$: In such a case, by our inductive hypothesis, there exists $\nu\tilde{r}.B \equiv A_{m-1}$ and a frame ϕ such that $S(\mathsf{tr}) \subseteq S(\mathsf{tr}')$ and $A \xRightarrow{\mathsf{tr}'}_{\mathsf{p}} \nu\tilde{r}.(B \mid \phi)$. W.l.o.g. we can assume that bound names and variables in $\tilde{r}.(B \mid \phi)$ are bound once and distinct from free names and variables. We can also assume that B is name-cleaned. We do a case analysis on the rule applied in $A_{m-1} \xrightarrow{\ell_m} A_m$.

- Case 1, any rule but the rule COMM: In such a case, by definition of the semantics, the result directly holds
- Case 2, rule COMM: In such a case, $B = \mathsf{in}^{\mathsf{ho}}(c, x).P_1 \mid \mathsf{out}^{\mathsf{ho}}(c, u).P_2 \mid P_3$ and $A_m = \nu\tilde{r}.(P_1\{^u/_x\} \mid P_2 \mid P_3)$. We do a case analysis on c and u:
  - $*$ $c \in \tilde{r}$: then since $c \notin oc(A)$ (A_{m-1} is honest) and by applying rule C-PRIV we obtain that $\tilde{r}.(B \mid \phi) \xRightarrow{\varepsilon}_{\mathsf{p}} \tilde{r}.(P_1\{^u/_x\} \mid P_2 \mid P_3 \mid \phi)$. Hence the result holds.
  - $*$ $c \notin \tilde{r}$ and u is of base type: By applying OUT-T followed by IN, we obtain that $\nu\tilde{r}.(B \mid \phi) \xrightarrow{\nu z.out(c,z).in(c,z)}_{\mathsf{p}} \nu\tilde{r}.(P_1\{^u/_x\} \mid P_2 \mid P_3 \mid \phi \mid \{^u/_z\})$ with z fresh. Hence the result holds.
  - $*$ $c \notin \tilde{r}$ and u is of channel type: By applying OUT-CH followed by IN, we obtain that $\nu\tilde{r}.(B \mid \phi) \xRightarrow{out(c,u).in(c,u)}_{\mathsf{p}} \nu\tilde{r}.(P_1\{^u/_x\} \mid P_2 \mid P_3 \mid \phi \mid \{^u/_x\})$. Hence the result holds.

We conclude by noticing that if $A \downdownarrows^{\mathsf{c}}_d$ then there exist A_c, tr_c such that $A \xRightarrow{\mathsf{tr}_c}_{\mathsf{c}} A_c$ and $d \in S(\mathsf{tr}_c)$. Thus by our property, we obtain that there exist A_p, tr_p such that $A \xRightarrow{\mathsf{tr}_p}_{\mathsf{p}} A_p$ and $S(\mathsf{tr}_c) \subseteq S(\mathsf{tr}_p)$ and so $d \in S(\mathsf{tr}_p)$ which implies $A \downdownarrows^{\mathsf{p}}_d$.

Implication 2, $A \downdownarrows^{\mathsf{p}}_d \Rightarrow A \downdownarrows^{\mathsf{e}}_d$: As $A \downdownarrows^{\mathsf{p}}_d$, there exists tr, A' such that $A \xRightarrow{\mathsf{tr}}_{\mathsf{p}} A'$ and $d \in S(\mathsf{tr})$. Since $\xrightarrow{\ell}_{\mathsf{p}} \subset \xrightarrow{\ell}_{\mathsf{e}}$, $A \xRightarrow{\mathsf{tr}}_{\mathsf{e}} A'$ and so $A \downdownarrows^{\mathsf{e}}_d$.

Implication 3, $A \downdownarrows^{\mathsf{e}}_d \Rightarrow A \downdownarrows^{\mathsf{c}}_d$: Since A is honest, the only rules that differ are the rules COMM, C-PRIV, EAV-OCH, EAV-CH, EAV-T.

We show that for all $A \xRightarrow{\mathsf{tr}}_{\mathsf{e}} A'$, there exist tr' such that $A \xRightarrow{\mathsf{tr}'}_{\mathsf{c}} A'$ and $S(\mathsf{tr}) \subseteq S(\mathsf{tr}')$. We prove this result by induction on the length of the derivation $A \xrightarrow{\ell_1\ldots\ell_m}_{\mathsf{c}} A'$ with tr being $\ell_1 \ldots \ell_m$ without the τ actions.

Base case $m = 0$: Hence $\mathsf{tr} = \varepsilon$ and so the result directly holds with $\mathsf{tr}' = \varepsilon$.

Inductive step $m > 0$: In such a case, by our inductive hypothesis, there exists tr'' such that $A \xRightarrow{\mathsf{tr}''}_{\mathsf{e}} A_{m-1}$. W.l.o.g. we can assume that bound names and variables in A_{m-1} are bound once and distinct from free names and variables. Moreover we can assume that $A_{m-1} = \nu\tilde{n}.B$ with B name-cleaned. We do a case analysis on the rule applied in $A_{m-1} \xrightarrow{\ell_m} A_m$.

- Case 1, rule C-PRIV: In such a case, $B = \mathsf{out}^{\mathsf{ho}}(c, u).P \mid \mathsf{in}^{\mathsf{ho}}(c, x).Q \mid R, c \in \tilde{n}$ and $A_m \equiv \nu\tilde{n}.(P \mid Q\{^u/_x\} \mid R)$. Notice that $B \xrightarrow{\tau}_{\mathsf{c}} \nu\tilde{n}.(P \mid Q\{^u/_x\} \mid R)$ by rule COMM hence the result holds with $\mathsf{tr}' = \mathsf{tr}''$.
- Case 2, rule EAV-OCH: In such a case, $B = \mathsf{out}^{\mathsf{ho}}(c, u).P \mid \mathsf{in}^{\mathsf{ho}}(c, x).Q \mid R$, $\ell = \nu u.eav(c, u)$, u is of channel type, $u \in \tilde{n}$ and $A_m \equiv \nu\tilde{n}'.(P \mid Q\{^u/_x\} \mid R)$ with $\tilde{n} = \tilde{n}'.u$. By applying rule OPEN-CH followed by rule IN, we obtain that $A_{m-1} \xRightarrow{\nu u.out(c,u).in(c,u)}_{\mathsf{c}} A_m$. Hence the result holds with $\mathsf{tr}' = \mathsf{tr}''.\nu u.out(c, u).in(c, u)$.

- Case 3, rule EAV-CH: In such a case, $B = \mathsf{out}^{\mathsf{ho}}(c, u).P \mid \mathsf{in}^{\mathsf{ho}}(c, x).Q \mid R$, $\ell = eav(c, u)$, u is of channel type, $u \notin \tilde{n}$ and $A_m \equiv \nu\tilde{n}.(P \mid Q\{^u/_x\} \mid R)$. By applying rule OUT-CH followed by rule IN, we obtain that $A_{m-1} \xRightarrow{out(c,u).in(c,u)}_{\mathsf{c}} A_m$. Hence the result holds with $\mathsf{tr}' = \mathsf{tr}''.out(c, u).in(c, u)$.
- Case 4, rule EAV-T: In such a case, $B = \mathsf{out}^{\mathsf{ho}}(c, u).P \mid \mathsf{in}^{\mathsf{ho}}(c, x).Q \mid R$, $\ell = \nu z.eav(c, z)$, u is of base type and $A_m \equiv \nu\tilde{n}.(P \mid Q\{^u/_x\} \mid R \mid \{^u/_z\})$. By applying rule OUT-T followed by rule IN, we obtain that $A_{m-1} \xRightarrow{\nu z.out(c,z).in(c,z)}_{\mathsf{c}} A_m$. Hence the result holds with $\mathsf{tr}' = \mathsf{tr}''.\nu z.out(c, z).in(c, z)$.
- Case 5, any other rule: In such a case, by definition of the semantics, the result directly holds.

We conclude by noticing that if $A \Downarrow^{\mathsf{e}}_{d}$ then there exist A', tr_e such that $A \overset{\mathsf{tr}_e}{\Rightarrow}_{\mathsf{e}} A'$ and $d \in S(\mathsf{tr}_e)$. Thus by our property, we obtain that there exist tr_c such that $A \overset{\mathsf{tr}_c}{\Rightarrow}_{\mathsf{c}} A'$ and $S(\mathsf{tr}_e) \subseteq S(\mathsf{tr}_c)$ and so $d \in S(\mathsf{tr}_c)$ which implies $A \Downarrow^{\mathsf{c}}_{d}$. □

Appendix F. Proof of Theorem 7

Lemma 13. *When restricted to $\mathcal{D}(\mathsf{p})$, we have $\approx^{\mathsf{p}}_{r} = \approx^{\mathsf{e}}_{r} \subsetneq \approx^{\mathsf{c}}_{r}$ for $r \in \{\ell, t\}$.*

Proof. By Lemma 3, we only need to consider the case $r = \ell$.

Before proving the main result, we show the following preliminary result: For all honest processes A, B, if $B \in \mathcal{D}(\mathsf{p})$, $A \approx^{\mathsf{p}}_{\ell} B$, $A \equiv \nu\tilde{n}.(\mathsf{out}^{\mathsf{ho}}(c, u).P \mid \mathsf{in}^{\mathsf{ho}}(c, x).Q \mid R)$ and $c \notin \tilde{n}$ then $B \overset{\varepsilon}{\Rightarrow}_{\mathsf{p}} B'$ with $B' = \nu\tilde{m}.(\mathsf{out}^{\mathsf{ho}}(c, v).P' \mid \mathsf{in}^{\mathsf{ho}}(c, x).Q' \mid R')$ and $\nu\tilde{n}.(\{^u/_z\} \cup \phi(R)) \sim \nu\tilde{m}.(\{^v/_z\} \cup \phi(R'))$ where z is fresh.

Note that $A \xrightarrow{in(c,a)} \nu\tilde{n}.(\mathsf{out}^{\mathsf{ho}}(c, u).P \mid Q\{^a/_x\} \mid R)$ for any a. As $A \approx^{\mathsf{p}}_{\ell} B$, there exists $B \overset{\varepsilon}{\Rightarrow}_{\mathsf{p}} B_1 \xrightarrow{in(c,a)}_{\mathsf{p}} B_1' \overset{\varepsilon}{\Rightarrow}_{\mathsf{p}} B_1''$ with $B_1 = \nu\tilde{m}.(in(c, x).Q' \mid R'')$. Since $B \in \mathcal{D}(\mathsf{p})$, we deduce that $B \approx^{\mathsf{p}}_{\ell} B_1$ and so $B_1 \approx^{\mathsf{p}}_{\ell} A$. But $A \xrightarrow{\nu z.out(c,z)}_{\mathsf{p}} A''$ for some A''. Thus, $B_1 \approx^{\mathsf{p}}_{\ell} A$ implies that there exists $B_1 \overset{\varepsilon}{\Rightarrow}_{\mathsf{p}} B_2 \xrightarrow{\nu z.out(c,z)}_{\mathsf{p}} B_2' \overset{\varepsilon}{\Rightarrow}_{\mathsf{p}} B_2''$ with $B_2 = \nu\tilde{k}.(in(c, x).Q' \mid out(c, v).P' \mid R')$ for some P', R' and $\phi(A'') \sim \phi(B_2'')$. Second, note that the subprocess $in(c, x).Q'$ in B_1 is also in B_2 since no internal communication can be applied on the public channel c. Hence we consider the result with $B' = B_2$. Second, notice that $\phi(B_2'') = \phi(B_2')$ and v is the term output in the transition $B_2 \xrightarrow{\nu z.out(c,z)}_{\mathsf{p}} B_2'$. Hence, we obtain $\nu\tilde{n}.(\{^u/_z\} \cup \phi(R)) \sim \nu\tilde{k}.(\{^v/_z\} \cup \phi(R'))$ for some fresh variable z. Thus the result holds.

Let us now focus on the main result. We start by proving $\approx^{\mathsf{p}}_{\ell} = \approx^{\mathsf{e}}_{\ell}$. By Theorem 5 we have that $\approx^{\mathsf{e}}_{\ell} \subseteq \approx^{\mathsf{p}}_{\ell}$. It remains to show that $\approx^{\mathsf{p}}_{\ell} \subseteq \approx^{\mathsf{e}}_{\ell}$.

Let $A, B \in \mathcal{D}(\mathsf{p})$ such that $A \approx^{\mathsf{p}}_{\ell} B$. We show that $\approx^{\mathsf{p}}_{\ell}$ is also a labelled bisimulation in the eavesdrop semantics.

- Since $A \approx^{\mathsf{p}}_{\ell} B$, we have that $\phi(A) \sim \phi(B)$.
- if $A \xrightarrow{\tau}_{\mathsf{e}} A'$ then, as B is a honest process, no C-EAV or C-OEAV transition is possible. Thus $A \xrightarrow{\tau}_{\mathsf{p}} A'$. As $A \approx^{\mathsf{p}}_{\ell} B$, there exists $B \overset{\varepsilon}{\Rightarrow}_{\mathsf{p}} B'$ such that $A' \approx^{\mathsf{p}}_{\ell} B'$. Since $\xrightarrow{\tau}_{\mathsf{p}} \subseteq \xrightarrow{\tau}_{\mathsf{e}}$, we have $B \overset{\varepsilon}{\Rightarrow}_{\mathsf{e}} B'$.
- if $A \xrightarrow{\ell}_{\mathsf{e}} A'$ with $\ell = \nu x.out(c, x)$ or $\ell = in(c, M)$ then $A \xrightarrow{\ell}_{\mathsf{p}} A'$. As $A \approx^{\mathsf{p}}_{\ell} B$, there exists $B \overset{\ell}{\Rightarrow}_{\mathsf{p}} B'$ such that $A' \approx^{\mathsf{p}}_{\ell} B'$. Once again since $\xrightarrow{\tau}_{\mathsf{p}} \subseteq \xrightarrow{\tau}_{\mathsf{e}}$, we have $B \overset{\ell}{\Rightarrow}_{\mathsf{e}} B'$.
- if $A \xrightarrow{\nu z.eav(c,z)}_{\mathsf{e}} A'$ then $A \equiv \nu\tilde{n}.(\mathsf{out}^{\mathsf{ho}}(c, u).P \mid \mathsf{in}^{\mathsf{ho}}(c, x).Q \mid R)$ and $A' = \nu\tilde{n}.(P \mid Q\{^u/_x\} \mid R \mid \{^u/_z\})$. Note that $A \xRightarrow{\nu z.out(c,z).in(c,z)}_{\mathsf{p}} A'$. As $A \approx^{\mathsf{p}}_{\ell} B$, there exists $B \xRightarrow{\nu z.out(c,z).in(c,z)}_{\mathsf{p}} B'$ and

$A' \approx^{\mathsf{p}}_{\ell} B'$. Thanks to our preliminary result, we know that there exists $B \overset{\varepsilon}{\Rightarrow}_{\mathsf{p}} B_2$ with $B_2 = \nu\tilde{m}.(\mathsf{out}^{\mathsf{ho}}(c,v).P' \mid \mathsf{in}^{\mathsf{ho}}(c,x).Q' \mid R')$ and $\nu\tilde{n}.(\{^{u}/_{z}\} \cup \phi(R)) \sim \nu\tilde{m}.(\{^{v}/_{z}\} \cup \phi(R'))$ where z is fresh. Thus, $B \overset{\varepsilon}{\Rightarrow}_{\mathsf{e}} B_2 \xrightarrow{\nu z.eav(c,z)}_{\mathsf{e}} B_3$, $\phi(B_3) \sim \phi(A')$ and $B \xRightarrow{\nu z.out(c,z).in(c,z)}_{\mathsf{p}} B_3$ for some B_3. Recall that $A' \approx^{\mathsf{p}}_{\ell} B'$ meaning that $\phi(A') \sim \phi(B')$ and so $\phi(B') \sim \phi(B_3)$. As $B \in \mathcal{D}(\mathsf{p})$, $\phi(B') \sim \phi(B_3)$, $B \xRightarrow{\nu z.out(c,z).in(c,z)}_{\mathsf{p}} B_3$ and $B \xRightarrow{\nu z.out(c,z).in(c,z)}_{\mathsf{p}} B'$ imply $B_3 \approx^{\mathsf{p}}_{\ell} B'$. As $A' \approx^{\mathsf{p}}_{\ell} B'$, $A' \approx^{\mathsf{p}}_{\ell} B_3$. Hence, we showed that $B \xRightarrow{\nu z.eav(c,z)}_{\mathsf{e}} B_3$ and $A' \approx^{\mathsf{p}}_{\ell} B_3$ which allows us to conclude the proof of $\approx^{\mathsf{p}}_{\ell} \subseteq \approx^{\mathsf{e}}_{\ell}$.

Let us now prove that $\approx^{\mathsf{p}}_{\ell} \subseteq \approx^{\mathsf{c}}_{\ell}$. We showed above that $\approx^{\mathsf{p}}_{\ell} = \approx^{\mathsf{e}}_{\ell}$. Moreover, we have that $\approx^{\mathsf{e}}_{\ell} \subseteq \approx^{\mathsf{c}}_{\ell}$ by Theorem 5, which allows us to conclude.

We conclude our proof by showing that the inclusion $\approx^{\mathsf{p}}_{\ell} \subseteq \approx^{\mathsf{c}}_{\ell}$ is strict. Consider the processes P and Q displayed in Fig. 9b. First notice that $P \not\approx^{\mathsf{p}}_{\ell} Q$ since $Q \xRightarrow{\nu x.out(d,x)}_{\mathsf{p}} Q'$ for some Q' but the process P cannot output on d directly. We can also easily show that $P, Q \in \mathcal{D}(\mathsf{p})$ and $P \approx^{\mathsf{c}}_{\ell} Q$. Indeed, first the frame of any transition consists only of multiple outputs of the public name a. Therefore the static equivalence always holds. Second, any action on P, i.e. output on c or d and input on c can always be matched on Q by unfolding a replication. Similarly, any output or input on c from Q can be matched P by unfolding a replication. Finally, any output on d from Q can be matched on P by unfolding the replications and applying an internal communication on c. Therefore $P \approx^{\mathsf{c}}_{\ell} Q$. The proof of $Q \in \mathcal{D}(\mathsf{p})$ is similar. To prove that $P \in \mathcal{D}(\mathsf{p})$, one must notice that in $P \overset{\mathsf{tr}}{\Rightarrow}_{\mathsf{p}} P'$, the number of output transitions on d available on P' is uniquely defined by tr. Thus, when considering $P \overset{\mathsf{tr}}{\Rightarrow}_{\mathsf{p}} P_1$ and $P \overset{\mathsf{tr}}{\Rightarrow}_{\mathsf{p}} P_2$, an output on d is possible on P_1 if and only if an output on d is possible on P_2. Hence $P_1 \approx^{\mathsf{p}}_{\ell} P_2$ and so $P \in \mathcal{D}(\mathsf{p})$. □

Lemma 4. *$\mathcal{D}(\mathsf{p}) = \mathcal{D}(\mathsf{e})$, $\mathcal{D}(\mathsf{c}) \not\subseteq \mathcal{D}(\mathsf{p})$ and $\mathcal{D}(\mathsf{p}) \not\subseteq \mathcal{D}(\mathsf{c})$.*

Proof. We start by showing that $\mathcal{D}(\mathsf{p}) \not\subseteq \mathcal{D}(\mathsf{c})$. Consider the process A displayed in Fig. 9a. $A \in \mathcal{D}(\mathsf{c})$ since $A \xrightarrow{\tau}_{\mathsf{c}} \mathsf{out}^{\mathsf{ho}}(c,a)$ by the rule COMM and $\mathsf{out}^{\mathsf{ho}}(c,a) \not\approx^{\mathsf{c}}_{\ell} A$. Moreover, $A \in \mathcal{D}(\mathsf{p})$ since for all tr, there is a unique A' such that $A \overset{\mathsf{tr}}{\Rightarrow}_{\mathsf{p}} A'$. Hence $\mathcal{D}(\mathsf{p}) \not\subseteq \mathcal{D}(\mathsf{c})$.

We now show that $\mathcal{D}(\mathsf{c}) \not\subseteq \mathcal{D}(\mathsf{p})$. Consider the process B displayed in Fig. 9b. Intuitively, the use of the private channel s in B encodes a non determinist choice between the two processes P and Q. We already showed in the proof of Lemma 13 that $P \not\approx^{\mathsf{p}}_{\ell} Q$ ands $P \approx^{\mathsf{c}}_{\ell} Q$. With a similar proof, we can also show that $P, Q \in \mathcal{D}(\mathsf{c})$ which allows us to deduce that $B \in \mathcal{D}(\mathsf{c})$. However, $B \overset{\varepsilon}{\Rightarrow}_{\mathsf{p}} P$, $B \overset{\varepsilon}{\Rightarrow}_{\mathsf{p}} Q$ and $P \not\approx^{\mathsf{p}}_{t} Q$ imply $B \notin \mathcal{D}(\mathsf{p})$.

Let us show $\mathcal{D}(\mathsf{e}) \subseteq \mathcal{D}(\mathsf{p})$. Consider an honest closed process A such that $A \in \mathcal{D}(\mathsf{e})$. Let $A \overset{tr}{\Rightarrow}_{\mathsf{p}} A_1$ and $A \overset{tr}{\Rightarrow}_{\mathsf{p}} A_2$. By definition of the semantics, $A \overset{tr}{\Rightarrow}_{\mathsf{p}} A_i$ implies $A \overset{tr}{\Rightarrow}_{\mathsf{e}} A_i$, for $i = 1, 2$. Since $A \in \mathcal{D}(\mathsf{e})$, we deduce $A_1 \approx^{\mathsf{e}}_{\ell} A_2$. By applying Theorem 5, we obtain $A_1 \approx^{\mathsf{p}}_{\ell} A_2$ which concludes the proof of $\mathcal{D}(\mathsf{e}) \subseteq \mathcal{D}(\mathsf{p})$.

Finally, let us show that $\mathcal{D}(\mathsf{p}) \subseteq \mathcal{D}(\mathsf{e})$. Let $A \in \mathcal{D}(\mathsf{p})$, $A \overset{\mathsf{tr}}{\Rightarrow}_{\mathsf{e}} A_1$ and $A \overset{\mathsf{tr}}{\Rightarrow}_{\mathsf{e}} A_2$. We can define tr' from tr by replacing all instances of $\nu z.eav(c,z)$ by $\nu z.out(c,z).in(c,z)$ to obtain $A \overset{\mathsf{tr}'}{\Rightarrow}_{\mathsf{p}} A_1$ and $A \overset{\mathsf{tr}'}{\Rightarrow}_{\mathsf{p}} A_2$. As $A \in \mathcal{D}(\mathsf{p})$, $A_1 \approx^{\mathsf{p}}_{\ell} A_2$. Moreover, by definition of $\mathcal{D}(\mathsf{p})$, $A \in \mathcal{D}(\mathsf{p})$ also implies that $A_1, A_2 \in \mathcal{D}(\mathsf{p})$. By Lemma 13, $A_1 \approx^{\mathsf{p}}_{\ell} A_2$ implies $A_1 \approx^{\mathsf{e}}_{\ell} A_2$ which allows us to conclude. □

The following theorem now follows directly from Lemmas 4 and 13.

Theorem 7. *When restricted to $\mathcal{D}(\mathsf{p})$, we have $\approx^{s_1}_{r_1} = \approx^{s_2}_{r_2} \subsetneq \approx^{\mathsf{c}}_{r_3}$ for $s_1, s_2 \in \{\mathsf{p}, \mathsf{e}\}$, $r_1, r_2, r_3 \in \{\ell, t\}$.*

Appendix G. Proof of Theorem 8

Theorem 8. *When restricted to $\mathcal{BD}(\mathsf{p})$, we have that $\approx^{\mathsf{p}}_{r} = \approx^{\mathsf{e}}_{r} \subsetneq \approx^{\mathsf{c}}_{r}$ for $r \in \{\ell, t\}$.*

Proof. Thanks to Lemma 13, we already know that $\approx^{\mathsf{p}}_{\ell} = \approx^{\mathsf{e}}_{\ell}$ and $\approx^{\mathsf{p}}_{\ell} \subseteq \approx^{\mathsf{c}}_{\ell}$. We will show in Lemma 15 the stronger result that the implication is strict for the class $\mathcal{AD}$ of action determinate processes which is a subset of $\mathcal{BD}(\mathsf{p})$ (Lemma 7). □

Appendix H. Proof of Lemma 7

Lemma 14. $\mathcal{AD} \subsetneq \mathcal{BD}(\mathsf{p})$.

Proof. Let $P \in \mathcal{AD}$. We will show that $P \in \mathcal{D}(\mathsf{p})$ which allows us to conclude as processes in $\mathcal{AD}$ do not use replication. Let us define the following transition rule $\xrightarrow{\mathit{if}}$ such that $P \xrightarrow{\mathit{if}} P'$ iff $P \xrightarrow{\tau}{}^{*}_{\mathsf{p}} P'$ with only application of the rules Then or Else, and these two rules cannot be applied on P'. Since $bn(P) \cap \mathcal{C}h = \emptyset$, we deduce that the rule C-Priv cannot be applied (all channels are public). Hence the only possible τ transitions are the applications of the rules Then and Else.

Notice that $P \xrightarrow{\mathit{if}} P'$ implies $P \approx^{\mathsf{p}}_{\ell} P'$. Moreover for all traces $P \overset{tr}{\Rightarrow} P'$, we can always *swap* the internal transitions in the trace so that they are always applied before visible actions when possible. Hence we can always obtain a trace of the form $P \xrightarrow{\mathit{if}} P_1 \xrightarrow{\ell_1}_{\mathsf{p}} P'_1 \xrightarrow{\mathit{if}} P_2 \xrightarrow{\ell_2}_{\mathsf{p}} \ldots \xrightarrow{\mathit{if}} P_n \xrightarrow{\ell_n}_{\mathsf{p}} P'_n$ such that $P'_n \approx^{\mathsf{p}}_{\ell} P'$ and $tr = \ell_1 \ldots \ell_n$.

To prove that $P \in \mathcal{D}(\mathsf{p})$, we show the following property: for all $P \xrightarrow{\mathit{if}} P_1 \xrightarrow{\ell_1}_{\mathsf{p}} P'_1 \xrightarrow{\mathit{if}} P_2 \xrightarrow{\ell_2}_{\mathsf{p}} \ldots \xrightarrow{\mathit{if}} P_n \xrightarrow{\ell_n}_{\mathsf{p}} P'_n$, for all $P \xrightarrow{\mathit{if}} Q_1 \xrightarrow{\ell_1}_{\mathsf{p}} Q'_1 \xrightarrow{\mathit{if}} Q_2 \xrightarrow{\ell_2}_{\mathsf{p}} \ldots \xrightarrow{\mathit{if}} Q_n \xrightarrow{\ell_n}_{\mathsf{p}} Q'_n$, we have $Q_i \equiv P_i$ and $Q'_i \equiv P'_i$ for all $i \in \{1, \ldots, n\}$. This property can be proved by induction on n. The base case $n = 0$ being trivial, we focus on the inductive case $n > 0$. In such a case, by applying our inductive hypothesis, we deduce that $Q'_{n-1} \equiv P'_{n-1}$. Note that $Q'_{n-1} \xrightarrow{\mathit{if}} Q_n$, $P'_{n-1} \xrightarrow{\mathit{if}} P_n$ and $Q'_{n-1} \equiv P'_{n-1}$ implies that $P'_{n-1} \xrightarrow{\mathit{if}} Q_n$. As mentioned in the previous paragraph, it entails $P_n \equiv Q_n$. Thus, it remains to show that $P'_n \equiv Q'_n$. If $\ell_n = in(c, t)$ then in such a case, $P'_n \equiv \nu\tilde{k}.(in(c, x).R \mid U)$ for some R and U. But by definition of an action determinate process, we know that U does not have an input on c at top-level, i.e. $U \not\equiv in(c, y).R' \mid V$. Hence, there is only one possible transition ℓ_n on P_n meaning that $P'_n \equiv Q'_n$. A similar reasoning holds for the case $\ell_n = \nu z.out(c, z)$.

To see that the inclusion is strict, simply observe that for the process $P \mathrel{\hat{=}} \mathsf{out}^{\mathsf{ho}}(c, a) \mid \mathsf{out}^{\mathsf{ho}}(c, a)$, we have $P \in \mathcal{D}(\mathsf{p})$, but $P \notin \mathcal{AD}$. □

Appendix I. Proof of Theorem 9

Lemma 15. *There exist $P, Q \in \mathcal{AD}$ such that $P \approx^{\mathsf{c}}_{\ell} Q$ but $P \not\approx^{\mathsf{p}}_{t} Q$.*

Proof. Consider the processes P and Q displayed in Fig. 10. Notice that an input transition with channel d in the private semantics is possible on P but not on Q. Therefore, we directly deduce that $P \not\approx_t^{\mathsf{p}} Q$. Let us now prove that $P \approx_\ell^{\mathsf{c}} Q$. We do an analysis on the transitions on P and Q.

- $P \xrightarrow{\tau}_{\mathsf{c}} P'$: Note that from P, the only possible τ action is an internal communication on the channel c. More specifically,

  $$P' = \nu k_1, \dots, k_7.(R_1(k_1) \mid \mathsf{in}^{\mathsf{ho}}(d, x_2).\mathsf{if}\ x_2 = k_2\ \mathsf{then}\ \mathsf{out}^{\mathsf{ho}}(c, k_3))$$

  We can also apply an internal communication on Q to obtain the same process, i.e., $Q \xrightarrow{\tau}_{\mathsf{c}} P'$. Note that we trivially have $P' \approx_\ell^{\mathsf{c}} P'$.
- $Q \xrightarrow{\tau}_{\mathsf{c}} Q'$: Once again the only possible τ action is an internal communication on the channel c. In fact, we have $Q' = P'$ where P' was defined in the previous case with $P \xrightarrow{\varepsilon}_{\mathsf{c}} P'$.
- $P \xrightarrow{\nu z.out(c,z)}_{\mathsf{c}} P'$: In such a case, $P' = \nu k_1, \dots, k_5.(\mathsf{in}^{\mathsf{ho}}(c, x_1).R_1(x_1) \mid \mathsf{in}^{\mathsf{ho}}(d, x_2).\mathsf{if}\ x_2 = k_2\ \mathsf{then}\ \mathsf{out}^{\mathsf{ho}}(c, k_3) \mid \{^{k_1}/_{z}\})$. Moreover, notice that $Q \xrightarrow{\nu z.out(c,z)}_{\mathsf{c}} P'$ too. Hence the result holds.
- $P \xrightarrow{in(c,t)}_{\mathsf{c}} P'$: Since $k_1, \dots, k_5$ are all bound and there is no frame in P, we know that $t \neq k_i$ for all $i = 1 \dots 5$. Thus, $P' = \nu k_1, \dots, k_5.(R_1(t) \mid \mathsf{out}^{\mathsf{ho}}(c, k_1) \mid \mathsf{in}^{\mathsf{ho}}(d, x_2).\mathsf{if}\ x_2 = k_2\ \mathsf{then}\ \mathsf{out}^{\mathsf{ho}}(c, k_3))$. We can make two observations on the process P': The first one being that the only possible transition on $R_1(t)$ is the execution of the conditional leading to the nil process. The second one being that in k_2 does not appear anymore in P' other than in the test $x_2 = k_2$. Therefore, k_2 cannot be deducible from P' and so we deduce that $P' \approx_\ell^{\mathsf{c}} \nu k_1.(\mathsf{out}^{\mathsf{ho}}(c, k_1) \mid \mathsf{in}^{\mathsf{ho}}(d, x_2))$.

  For sake of readability, we denote by $\xrightarrow{\tau(c)}_{\mathsf{c}}$ a τ action corresponding to an internal communication on a channel c. Moreover, we denote by $\xrightarrow{\mathsf{if}}_{\mathsf{c}}$ a τ action corresponding to a conditional (i.e. rules Then or Else) and we denote by $\xRightarrow{\tau(c)}_{\mathsf{c}}$ the transition $\xrightarrow{\mathsf{if}}{}_{\mathsf{c}}^{*}\xrightarrow{\tau(c)}_{\mathsf{c}}\xrightarrow{\mathsf{if}}{}_{\mathsf{c}}^{*}$. Finally, we denote by $\tilde{k} = k_1, \dots, k_5$. Let us execute Q as follows:

  $$\begin{aligned} Q &\xrightarrow{\tau(c)}_{\mathsf{c}} \nu\tilde{k}.(R_1(k_1) \mid \mathsf{in}^{\mathsf{ho}}(d, x_2).\mathsf{if}\ x_2 = k_2\ \mathsf{then}\ \mathsf{out}^{\mathsf{ho}}(c, k_3)) \\ &\xRightarrow{\tau(d)}_{\mathsf{c}} \nu\tilde{k}.(\mathsf{in}^{\mathsf{ho}}(c, x_3).\mathsf{if}\ x_3 = k_3\ \mathsf{then}\ R_3\ \mathsf{else}\ \mathsf{in}^{\mathsf{ho}}(d, x) \mid \mathsf{out}^{\mathsf{ho}}(c, k_3)) \\ &\xrightarrow{in(c,t)}_{\mathsf{c}}\xrightarrow{\mathsf{if}}_{\mathsf{c}} \nu\tilde{k}.(\mathsf{in}^{\mathsf{ho}}(d, x) \mid \mathsf{out}^{\mathsf{ho}}(c, k_3)) \end{aligned}$$

  Let us denote $Q' = \tilde{k}.(\mathsf{in}^{\mathsf{ho}}(d, x) \mid \mathsf{out}^{\mathsf{ho}}(c, k_3))$. We trivially have that $Q' \approx_\ell^{\mathsf{c}} \nu k_1.(\mathsf{out}^{\mathsf{ho}}(c, k_1) \mid \mathsf{in}^{\mathsf{ho}}(d, x_2))$ and so $P' \approx_\ell^{\mathsf{c}} Q'$.
- $Q \xrightarrow{in(c,t)}_{\mathsf{c}} Q'$: Once again, we know that $t \neq k_i$ for all $i = 1 \dots 5$. Thus $Q' = \nu\tilde{k}.(R_1(t) \mid \mathsf{out}^{\mathsf{ho}}(c, k_1).\mathsf{in}^{\mathsf{ho}}(d, x_2).\mathsf{if}\ x_2 = k_2\ \mathsf{then}\ \mathsf{out}^{\mathsf{ho}}(c, k_3))$. With $t \neq k_1$ and the fact that k_2 is no longer deducible in Q', we obtain that $Q' \approx_\ell^{\mathsf{c}} \nu k_1.\mathsf{out}^{\mathsf{ho}}(c, k_1).\mathsf{in}^{\mathsf{ho}}(d, x_2)$.

  Let us execute P as follows:

  $$\begin{aligned} P &\xrightarrow{\tau(c)}_{\mathsf{c}} \nu\tilde{k}.(R_1(k_1) \mid \mathsf{in}^{\mathsf{ho}}(d, x_2).\mathsf{if}\ x_2 = k_2\ \mathsf{then}\ \mathsf{out}^{\mathsf{ho}}(c, k_3)) \\ &\xRightarrow{\tau(d)}_{\mathsf{c}} \nu\tilde{k}.(\mathsf{in}^{\mathsf{ho}}(c, x_3).\mathsf{if}\ x_3 = k_3\ \mathsf{then}\ R_3\ \mathsf{else}\ \mathsf{in}^{\mathsf{ho}}(d, x) \mid \mathsf{out}^{\mathsf{ho}}(c, k_3)) \\ &\xRightarrow{\tau(c)}_{\mathsf{c}} \nu\tilde{k}.R_3 \\ &\xrightarrow{in(c,t)}_{\mathsf{c}}\xrightarrow{\mathsf{if}}_{\mathsf{c}} \nu\tilde{k}.(\mathsf{out}^{\mathsf{ho}}(c, k_4).\mathsf{in}^{\mathsf{ho}}(d, x_5).R_5(x_5)) \end{aligned}$$

Let us denote $P' = \nu\tilde{k}.(\mathsf{out}^{\mathsf{ho}}(c, k_4).\mathsf{in}^{\mathsf{ho}}(d, x_5).R_5(x_5))$. Note that k_5 is not deducible in P' meaning that $P' \approx^{\mathsf{c}}_{\ell} \nu k_4.\mathsf{out}^{\mathsf{ho}}(c, k_4).\mathsf{in}^{\mathsf{ho}}(d, x_5)$. Since we already showed that $Q' \approx^{\mathsf{c}}_{\ell} \nu k_1.\mathsf{out}^{\mathsf{ho}}(c, k_1).\mathsf{in}^{\mathsf{ho}}(d, x_2)$, we conclude that $P' \approx^{\mathsf{c}}_{\ell} Q'$.

- $P \xrightarrow{in(d,t)}_{\mathsf{c}} P'$: In such a case, we have $P' = \nu\tilde{k}.(\mathsf{in}^{\mathsf{ho}}(c, x_1).R_1(x_1) \mid \mathsf{out}^{\mathsf{ho}}(c, k_1) \mid \mathsf{if}\ t = k_2\ \mathsf{then}\ \mathsf{out}^{\mathsf{ho}}(c, k_3))$. Note that $t \neq k_2$ and that so k_3 is not deducible in P'. Thus, we obtain the following:

  $$P' \approx^{\mathsf{c}}_{\ell} \nu\tilde{k}.(\mathsf{in}^{\mathsf{ho}}(c, x_1).\mathsf{if}\ x_1 = k_1\ \mathsf{then}\ \mathsf{out}^{\mathsf{ho}}(d, k_2).\mathsf{in}^{\mathsf{ho}}(c, x_3).\mathsf{in}^{\mathsf{ho}}(d, x) \mid \mathsf{out}^{\mathsf{ho}}(c, k_1))$$

  Let us execute Q as follows:

  $$\begin{array}{lll} Q & \xrightarrow{\tau(c)}_{\mathsf{c}} & \nu\tilde{k}.(R_1(k_1) \mid \mathsf{in}^{\mathsf{ho}}(d, x_2).\mathsf{if}\ x_2 = k_2\ \mathsf{then}\ \mathsf{out}^{\mathsf{ho}}(c, k_3)) \\ & \xRightarrow{\tau(d)}_{\mathsf{c}} & \nu\tilde{k}.(\mathsf{in}^{\mathsf{ho}}(c, x_3).\mathsf{if}\ x_3 = k_3\ \mathsf{then}\ R_3\ \mathsf{else}\ \mathsf{in}^{\mathsf{ho}}(d, x) \mid \mathsf{out}^{\mathsf{ho}}(c, k_3)) \\ & \xRightarrow{\tau(c)}_{\mathsf{c}} & \nu\tilde{k}.R_3 \\ & \xRightarrow{\tau(c)}_{\mathsf{c}} & \nu\tilde{k}.(\mathsf{in}^{\mathsf{ho}}(d, x_5).R_5(x_5) \mid \mathsf{out}^{\mathsf{ho}}(d, k_5)) \\ & \xrightarrow{\tau(d)}_{\mathsf{c}} & \nu\tilde{k}.R_5(k_5) \\ & \xrightarrow{\mathsf{if}}_{\mathsf{c}}\xrightarrow{in(c,t)}_{\mathsf{c}} & \nu\tilde{k}.(\mathsf{in}^{\mathsf{ho}}(c, x_6).\mathsf{if}\ x_6 = k_6\ \mathsf{then}\ \mathsf{out}^{\mathsf{ho}}(d, k_7).\mathsf{in}^{\mathsf{ho}}(c, x_3).\mathsf{in}^{\mathsf{ho}}(d, x) \\ & & \quad \mid \mathsf{out}^{\mathsf{ho}}(c, k_6)) \end{array}$$

  As we already proved $P' \approx^{\mathsf{c}}_{\ell} \nu\tilde{k}.(\mathsf{in}^{\mathsf{ho}}(c, x_1).\mathsf{if}\ x_1 = k_1\ \mathsf{then}\ \mathsf{out}^{\mathsf{ho}}(d, k_2).\mathsf{in}^{\mathsf{ho}}(c, x_3).\mathsf{in}^{\mathsf{ho}}(d, x) \mid \mathsf{out}^{\mathsf{ho}}(c, k_1))$, we conclude that $P' \approx^{\mathsf{c}}_{\ell} Q'$.

We conclude that $P \approx^{\mathsf{c}}_{\ell} Q$.

It remains to prove that $P, Q \in \mathcal{D}(\mathsf{p})$. We show in fact that P and Q are both action-determinate and we conclude by applying Lemma 7.

We do a case analysis on the trace of P.

Case 1: $P \xrightarrow{in(c,t)}_{\mathsf{c}} P_1$. In such a case,

$$P_1 = \nu\tilde{k}.(R_1(t) \mid \mathsf{out}^{\mathsf{ho}}(c, k_1) \mid \mathsf{in}^{\mathsf{ho}}(d, x_2).\mathsf{if}\ x_2 = k_2\ \mathsf{then}\ \mathsf{out}^{\mathsf{ho}}(c, k_3))$$

Since $t \neq k_1$, $R_1(t)$ can only be reduced into the nil process. Moreover, k_2 is not deducible in P_1, meaning that $\mathsf{out}^{\mathsf{ho}}(c, k_3)$ can never be executed. Thus, P_1 is action-determinate.

Case 2: $P \xrightarrow{in(d,t)}_{\mathsf{c}} P_1$. In such a case,

$$P_1 = \nu\tilde{k}.(\mathsf{in}^{\mathsf{ho}}(c, x_1).R_1(x_1) \mid \mathsf{out}^{\mathsf{ho}}(c, k_1) \mid \mathsf{if}\ t = k_2\ \mathsf{then}\ \mathsf{out}^{\mathsf{ho}}(c, k_3))$$

Since $t \neq k_2$, $\mathsf{out}^{\mathsf{ho}}(c, k_3)$ can never be executed meaning that k_3 is not deducible in P_1. As such, we deduce that proving that P_1 is action-determinate is equivalent to proving that the following process is action-determinate:

$$\nu\tilde{k}.(\mathsf{in}^{\mathsf{ho}}(c, x_1).\mathsf{if}\ x_1 = k_1\ \mathsf{then}\ \mathsf{out}^{\mathsf{ho}}(d, k_2).\mathsf{in}^{\mathsf{ho}}(c, x_3).\mathsf{in}^{\mathsf{ho}}(d, x) \mid \mathsf{out}^{\mathsf{ho}}(c, k_1))$$

Since there is no common actions between the two parallel processes in the previous process, we conclude that it is action-determinate.

Case 3: $P \xrightarrow{\nu z.out(c,z)}_{\mathsf{c}} P_1$. In such a case,

$$P_1 = \nu\tilde{k}.(\mathsf{in}^{\mathsf{ho}}(c, x_1).R_1(x_1) \mid \mathsf{in}^{\mathsf{ho}}(d, x_2).\mathsf{if}\ x_2 = k_2\ \mathsf{then}\ \mathsf{out}^{\mathsf{ho}}(c, k_3) \mid \{^{k_1}/_{z}\})$$

We have to consider two possible actions next, that are either an input on c or an input on d. If we consider $P_1 \xrightarrow{in(d,t)}_{\mathsf{c}} P_2$ then once again, $t \neq k_2$ meaning that $\mathsf{out}^{\mathsf{ho}}(c, k_3)$ can never be executed. Thus proving P_2 is action-determinate is equivalent to proving that the following process is action-determinate:

$$\nu\tilde{k}.(\mathsf{in}^{\mathsf{ho}}(c, x_1).\mathsf{if}\ x_1 = k_1\ \mathsf{then}\ \mathsf{out}^{\mathsf{ho}}(d, k_2).\mathsf{in}^{\mathsf{ho}}(c, x_3).\mathsf{in}^{\mathsf{ho}}(d, x) \mid \{^{k_1}/_{z}\})$$

This process is trivially action-determinate. Thus, let us consider $P_1 \xrightarrow{in(c,t)} P_2$, i.e.

$$P_2 = \nu\tilde{k}.(R_1(t\{^{k_1}/_{z}\}) \mid \mathsf{in}^{\mathsf{ho}}(d, x_2).\mathsf{if}\ x_2 = k_2\ \mathsf{then}\ \mathsf{out}^{\mathsf{ho}}(c, k_3) \mid \{^{k_1}/_{z}\})$$

If $t \neq z$, i.e. $t\{^{k_1}/_{z}\} \neq k_1$, then $R_1(t\{^{k_1}/_{z}\})$ can only be reduced to the nil process and so P_2 would be trivially action-determinate. Thus, let us assume that $t = z$.

Note that the determinacy of P_2 can only be broken if R_3 can be executed but after the following transitions $P_2 \xrightarrow{\nu z_2.out(d,z_2)}_{\mathsf{c}} \xrightarrow{in(d,z_2)}_{\mathsf{c}} \xrightarrow{\nu z_3.out(c,z_3)}_{\mathsf{c}} \xrightarrow{in(c,z_3)}_{\mathsf{c}} P_3 = \nu\tilde{k}.(R_3 \mid \{^{k_1}/_{z} \,|^{k_2}\, /_{z_2} \,|^{k_3}\, /_{z_3}\})$.

Therefore, we only need to show that R_3 is action-determinate. Once again, this may not be the case only if $R_5(k_5)$ is executed. However, this is only possible with transitions as follows: $P_3 \xrightarrow{\nu z_4 out(c,z_4)}_{\mathsf{c}} \xrightarrow{in(c,z_4)}_{\mathsf{c}} \xrightarrow{\nu z_3.out(d,z_5)}_{\mathsf{c}} \xrightarrow{in(d,z_5)}_{\mathsf{c}} \nu\tilde{k}.R_5(k_5)$. Since $R_5(k_5)$ is trivially action-determinate, we conclude that P is an action-determinate process.

The proof of Q being action-determinate is similar. □

Theorem 9. *When restricted to $\mathcal{AD}$, we have that $\approx^{\mathsf{p}}_{r} = \approx^{\mathsf{e}}_{r} \subsetneq \approx^{\mathsf{c}}_{r}$ for $r \in \{\ell, t\}$.*

Proof. Thanks to Lemmas 13 and 7, we already know that $\approx^{\mathsf{p}}_{\ell} = \approx^{\mathsf{e}}_{\ell}$ and $\approx^{\mathsf{p}}_{\ell} \subseteq \approx^{\mathsf{c}}_{\ell}$. The fact that the implication is strict follows from Lemma 15. □

Appendix J. Proof of Theorem 10

Lemma 16. *Let $P, Q \in \mathcal{SAD}$ such that $P \approx^{\mathsf{c}}_{\ell} Q$. If there exists c such that $P \xRightarrow{\nu x.out(c,x)}_{\mathsf{p}} P_1$ then there exists d such that:*

- $P \xRightarrow{\nu x.out(d,x)}_{\mathsf{p}} P'$
- $Q \xRightarrow{\nu x.out(d,x)}_{\mathsf{p}} Q'$
- $P' \approx^{\mathsf{c}}_{\ell} Q'$.

Proof. Consider the maximal trace $P \xRightarrow{\nu x.out(d,x)}_{\mathsf{c}} P_2 \xRightarrow{tr}_{\mathsf{c}} P_3$ of P that starts with an output. Note that in fact $P \xRightarrow{\nu x.out(d,x)}_{\mathsf{p}} P_2$. Indeed, if $P \xRightarrow{\nu x.out(d,x)}_{\mathsf{c}} P_2$ contains an internal communication transition, the we can transform this transition into two transitions $\xrightarrow{\nu z.out(c,z)}_{\mathsf{c}} \xrightarrow{in(c,z)}_{\mathsf{c}}$ which would contradicts the maximality hypothesis on $P \xRightarrow{\nu x.out(d,x)}_{\mathsf{c}} P_2 \xRightarrow{tr}_{\mathsf{c}} P_3$. Since $P \approx^{\mathsf{c}}_{\ell} Q$, we know that $Q \xRightarrow{\nu x.out(d,x)}_{\mathsf{c}} Q_2 \xRightarrow{tr}_{\mathsf{c}} Q_3$ with

$P_2 \approx^{\mathsf{c}}_{\ell} Q_2$ and $P_3 \approx^{\mathsf{c}}_{\ell} Q_3$. If $Q \xRightarrow{\nu x.out(d,x)}_{\mathsf{p}} Q_2$ then the result holds. Otherwise, there is an internal communication transition on some channel c in $Q \xRightarrow{\nu x.out(d,x)}_{\mathsf{c}} Q_2$. Once again, we can replace this transition into two transitions $\xrightarrow{\nu z.out(c,z)}_{\mathsf{c}}\xrightarrow{in(c,z)}_{\mathsf{c}}$ with $\mathsf{tr}' = \nu z.out(c, z).in(c, z)$, yielding $Q \overset{\mathsf{tr}'}{\Rightarrow}_{\mathsf{c}}\xRightarrow{\nu x.out(d,x)}_{\mathsf{c}}\overset{\mathsf{tr}}{\Rightarrow}_{\mathsf{c}} Q'_3$ or $Q \xRightarrow{\nu x.out(d,x)}_{\mathsf{c}}\overset{\mathsf{tr}'}{\Rightarrow}_{\mathsf{c}}\overset{\mathsf{tr}}{\Rightarrow}_{\mathsf{c}} Q'_3$ for some Q'_3. In both case, $\mathsf{tr}'.\nu x.out(d, x).\mathsf{tr}$ and $\nu x.out(d, x).\mathsf{tr}'.\mathsf{tr}$ start with an output. As $P \approx^{\mathsf{c}}_{\ell} Q$, $\mathsf{tr}'.\nu x.out(d, x).\mathsf{tr}$ or $\nu x.out(d, x).\mathsf{tr}'.\mathsf{tr}$ is also a trace of P which contradicts the maximality of $\nu x.out(d, x).\mathsf{tr}$. □

Lemma 17. *Let $P, Q \in \mathcal{SAD}$ such that $P \approx^{\mathsf{c}}_{\ell} Q$. If*

- *for all d, P', $P \not\xRightarrow{\nu x.out(d,x)}_{\mathsf{p}} P'$*
- *for all d, Q', $Q \not\xRightarrow{\nu x.out(d,x)}_{\mathsf{p}} Q'$*

then for all c, M, for all $P \xRightarrow{in(c,M)}_{\mathsf{p}} P'$, there exists $Q \xRightarrow{in(c,M)}_{\mathsf{p}} Q'$ such that $P' \approx^{\mathsf{c}}_{\ell} Q'$.

Proof. Since $P \xRightarrow{in(c,M)}_{\mathsf{p}} P'$ and $P \approx^{\mathsf{c}}_{\ell} Q$ then there exists $Q \xRightarrow{in(c,M)}_{\mathsf{c}} Q'$ and $P' \approx^{\mathsf{c}}_{\ell} Q'$. Thanks to our hypothesis on Q we know that $Q \xRightarrow{in(c,M)}_{\mathsf{p}} Q'' \xrightarrow{\tau}^{*}_{\mathsf{c}} Q'$ (No internal communication are possible directly on Q since not output is available). But $P \approx^{\mathsf{c}}_{\ell} Q$ and $Q \xRightarrow{in(c,M)}_{\mathsf{p}} Q''$ implies that $P \xRightarrow{in(c,M)}_{\mathsf{c}} P''$ with $P'' \approx^{\mathsf{c}}_{\ell} Q''$. Once again by our hypothesis on P, we have that $P \xRightarrow{in(c,M)}_{\mathsf{p}} P''' \xrightarrow{\tau}^{*}_{\mathsf{c}} P''$. As $P \in \mathcal{SAD}$, $P \in \mathcal{D}(\mathsf{p})$. As $P \xRightarrow{in(c,M)}_{\mathsf{p}} P'''$ and $P \xRightarrow{in(c,M)}_{\mathsf{p}} P'$, we deduce $P''' \approx^{\mathsf{p}}_{\ell} P'$ which implies $P''' \approx^{\mathsf{c}}_{\ell} P'$ by Lemma 13. To summarize, we have:

- $P'' \approx^{\mathsf{c}}_{\ell} Q''$
- $P' \approx^{\mathsf{c}}_{\ell} Q'$
- $Q'' \xrightarrow{\varepsilon}^{*}_{\mathsf{c}} Q'$
- $P''' \xrightarrow{\varepsilon}^{*}_{\mathsf{c}} P''$
- $P''' \approx^{\mathsf{c}}_{\ell} P'$

If in fact $Q'' \xrightarrow{\varepsilon}^{*}_{\mathsf{p}} Q'$ then the result directly holds. Else there exists an internal communication transition in $Q'' \xrightarrow{\varepsilon}^{*}_{\mathsf{c}} Q'$. Hence, $Q'' \xRightarrow{\nu z.out(d,z).in(d,z)}_{\mathsf{p}} Q' \mid \phi$ for some ϕ.

Take $Q' \overset{\mathsf{tr}}{\Rightarrow}_{\mathsf{c}} Q_1$ a maximal trace of Q', we deduce that $\nu z.out(d, z).in(d, z).\mathsf{tr}$ is a trace of Q''. Since $P'' \approx^{\mathsf{c}}_{\ell} Q''$, we know that $\nu z.out(d, z).in(d, z).\mathsf{tr}$ is a trace of P''. With $P''' \xrightarrow{\varepsilon}^{*}_{\mathsf{c}} P''$, we deduce that $\nu z.out(d, z).in(d, z).\mathsf{tr}$ is a trace of P'''. As $P''' \approx^{\mathsf{c}}_{\ell} P' \approx^{\mathsf{c}}_{\ell} Q'$, $\nu z.out(d, z).in(d, z).\mathsf{tr}$ is also a trace of Q' which gives a contradiction with tr being a maximal trace of Q'. □

Lemma 18. *Let $P, Q \in \mathcal{SAD}$ such that $P \approx^{\mathsf{c}}_{\ell} Q$. If there exists c such that*

- *$P \xrightarrow{\nu x.out(c,x)}_{\mathsf{p}} P'$,*
- *$Q \xrightarrow{\nu x.out(c,x)}_{\mathsf{p}} Q'$, and*
- *$P' \approx^{\mathsf{p}}_{\ell} Q'$*

then for all d, $P \xrightarrow{\nu y.out(d,y)}_{\mathsf{p}} P_1$ implies $Q \xRightarrow{\nu y.out(d,y)}_{\mathsf{p}} Q_1$ and $P_1 \approx^{\mathsf{c}}_{\ell} Q_1$.

Proof. Consider $P \xrightarrow{\nu y.out(d,y)}_{\mathsf{p}} P_1$. If $c = d$ then P being strongly action determinate implies that $P_1 = P'$ and so the result trivially holds. Therefore, let us consider that $c \neq d$.

In such a case, we deduce $P = \nu\tilde{n}.(\mathsf{out}^{\mathsf{ho}}(c,u).R_1 \mid \mathsf{out}^{\mathsf{ho}}(d,v).R_2 \mid R_3)$ and $Q = \nu\tilde{n}'.(\mathsf{out}^{\mathsf{ho}}(c,u').S_1 \mid S)$. Since $P \xrightarrow{\nu x.out(c,x)}_{\mathsf{p}} P'$ and $Q \xrightarrow{\nu x.out(c,x)}_{\mathsf{p}} Q'$, we deduce that $P' = \nu\tilde{n}.(R_1 \mid \mathsf{out}^{\mathsf{ho}}(d,v).R_2 \mid R_3 \mid \{^{u}/_{x}\})$ and $Q' = \nu\tilde{n}'.(S_1 \mid S \mid \{^{u'}/_{x}\})$. Note that $P' \approx_{\ell}^{\mathsf{p}} Q'$ implies that one of the following two cases:

(1) $S = \mathsf{out}^{\mathsf{ho}}(d,v').S_2 \mid S_3$
(2) $S_1 = \mathsf{out}^{\mathsf{ho}}(d,v').S_2 \mid S_3$

As $P' \approx_{\ell}^{\mathsf{p}} Q'$, we know that $P \xrightarrow{\nu x.out(c,x)}_{\mathsf{p}}\xrightarrow{\nu y.out(d,y)}_{\mathsf{p}} \nu\tilde{n}.P_2$ and $Q \xrightarrow{\nu x.out(c,x)}_{\mathsf{p}}\xRightarrow{\nu y.out(d,y)}_{\mathsf{p}} \nu\tilde{n}'.Q_2$ where

- $P_2 = R_1 \mid R_2 \mid R_3 \mid \{^{u}/_{x};^{v}/_{y}\}$
- $Q_2 = S_1 \mid S_2 \mid S_3 \mid \{^{u'}/_{x};^{v'}/_{y}\}$
- $\nu\tilde{n}.P_2 \approx_{\ell}^{\mathsf{p}} \nu\tilde{n}'.Q_2$.

Moreover, by $P \approx_{\ell}^{\mathsf{c}} Q$, we deduce that $Q \xRightarrow{\nu y.out(d,y)}_{\mathsf{c}} Q_4 \xRightarrow{\nu x.out(c,x)}_{\mathsf{c}} \nu\tilde{n}'.Q_3$ with $\nu\tilde{n}.P_2 \approx_{\ell}^{\mathsf{c}} \nu\tilde{n}'.Q_3$ and $P_1 \approx_{\ell}^{\mathsf{c}} Q_4$. If $Q \xRightarrow{\nu y.out(d,y)}_{\mathsf{p}} Q_4$ then the result trivially holds. Therefore, let us for now assume that $Q \xRightarrow{\nu y.out(d,y)}_{\mathsf{c}} Q_4$ contains some internal communication transitions.

In both cases 1 and 2, since Q is strongly action determinate, the outputs $\mathsf{out}^{\mathsf{ho}}(c,u')$ and $\mathsf{out}^{\mathsf{ho}}(d,v')$ in Q are necessarily executed either within an internal communication or with the output rule in the transition $Q \xRightarrow{\nu y.out(d,y)}_{\mathsf{c}} Q_4 \xRightarrow{\nu x.out(c,x)}_{\mathsf{c}} \nu\tilde{n}'.Q_3$. In both case, we can make apparent the internal communications in $Q \xRightarrow{\nu y.out(d,y)}_{\mathsf{c}}\xRightarrow{\nu x.out(c,x)}_{\mathsf{c}} \nu\tilde{n}'.Q_3$ and output first $\mathsf{out}^{\mathsf{ho}}(d,v')$ and $\mathsf{out}^{\mathsf{ho}}(c,u')$ giving us the following trace:

$$Q \xrightarrow{\nu x'.out(c,x')}_{\mathsf{p}}\xRightarrow{\nu y'.out(d,y')}_{\mathsf{p}} \nu\tilde{n}'.Q_2' \overset{tr}{\Rightarrow}_{\mathsf{c}} \nu\tilde{n}'(Q_3 \mid \Phi)$$

for some Φ, x', y' such that $\nu y.out(d,y)$ and $\nu x.out(c,x)$ are included in $\nu y'.out(d,y').\nu x'.out(c,x').tr$ and $Q_2' = S_1 \mid S_2 \mid S_3 \mid \{^{u'}/_{x'};^{v'}/_{y'}\}$

Since we assumed that $Q \xRightarrow{\nu y.out(d,y)}_{\mathsf{c}} Q_4$ contains some internal communications, we deduce that $tr \neq \varepsilon$. To summarized, we showed that:

- $\nu\tilde{n}.P_2 \approx_{\ell}^{\mathsf{p}} \nu\tilde{n}'.Q_2$
- $\nu\tilde{n}.P_2 \approx_{\ell}^{\mathsf{c}} \nu\tilde{n}'.Q_3$
- $\nu\tilde{n}'.Q_2' \overset{tr}{\Rightarrow}_{\mathsf{c}} \nu\tilde{n}'(Q_3 \mid \Phi)$
- $tr \neq \varepsilon$
- Q_2' is the process Q_2 where x and y in the domain of the frame have been replaced by x' and y' respectively.

Consider now a maximal trace of $\nu\tilde{n}'.Q_3$, i.e. $\nu\tilde{n}'.Q_3 \overset{tr'}{\Rightarrow}_{\mathsf{c}} Q_5$. Hence, tr' is a trace of $\nu\tilde{n}'.(Q_3 \mid \Phi)$. Hence, $tr.tr'$ is a trace of $\nu\tilde{n}'.Q_2'$. Thus, $tr.tr'\{^{x}/_{x'};\ ^{y}/_{y'}\}$ is a trace of $\nu\tilde{n}'.Q_2$. As $\nu\tilde{n}.P_2 \approx_{\ell}^{\mathsf{p}} \nu\tilde{n}'.Q_2$ implies $\nu\tilde{n}.P_2 \approx_{\ell}^{\mathsf{c}} \nu\tilde{n}'.Q_2$ (Lemma 13), $tr.tr'\{^{x}/_{x'};\ ^{y}/_{y'}\}$ is a trace of $\nu\tilde{n}.P_2$ and so a trace of $\nu\tilde{n}'.Q_3$ thanks to $\nu\tilde{n}.P_2 \approx_{\ell}^{\mathsf{c}} \nu\tilde{n}'.Q_3$. Since we assumed that $tr \neq \emptyset$, we obtain a contradiction with tr' being a maximal trace of $\nu\tilde{n}'.Q_3$. □

Corollary 4. *Let $P, Q \in \mathcal{SAD}$ such that $P \approx_\ell^{\mathsf{c}} Q$. Assume that for all P', Q', $|P'| + |Q'| < |P| + |Q|$ and $P' \approx_\ell^{\mathsf{c}} Q'$ implies $P' \approx_\ell^{\mathsf{p}} Q'$.*

For all d, $P \xrightarrow{\nu y.out(d,y)}_{\mathsf{p}} P_1$ implies $Q \xRightarrow{\nu y.out(d,y)}_{\mathsf{p}} Q_1$ and $P_1 \approx_p Q_1$.

Proof. Let $P \xrightarrow{\nu y.out(d,y)}_{\mathsf{p}} P_0$. By Lemma 16, we know that there exists c such that $P \xRightarrow{\nu x.out(c,x)}_{\mathsf{p}} P'$, $Q \xRightarrow{\nu x.out(c,x)}_{\mathsf{p}} Q'$ and $P' \approx_c Q'$. Since $|P'| + |Q'| < |P| + |Q|$, we deduce that $P' \approx_p Q'$. Note that $P, Q \in \mathcal{D}(\mathsf{p})$. Hence, there exists P_1, P_2, Q_1, Q_2 such that $P \overset{\varepsilon}{\Rightarrow}_{\mathsf{p}} P_1 \xrightarrow{\nu x.out(c,x)}_{\mathsf{p}} P_2 \overset{\varepsilon}{\Rightarrow}_{\mathsf{p}} P'$ and $Q \overset{\varepsilon}{\Rightarrow}_{\mathsf{p}} Q_1 \xrightarrow{\nu x.out(c,x)}_{\mathsf{p}} Q_2 \overset{\varepsilon}{\Rightarrow}_{\mathsf{p}} Q'$ such that $P \approx_\ell^{\mathsf{p}} P_1$, $P_2 \approx_\ell^{\mathsf{p}} P'$, $Q \approx_\ell^{\mathsf{p}} Q_1$ and $Q_2 \approx_\ell^{\mathsf{p}} Q'$. Note that by Lemma 13, $P \approx_\ell^{\mathsf{p}} P_1$ and $Q \approx_\ell^{\mathsf{p}} Q_1$ implies $P \approx_\ell^{\mathsf{c}} P_1$ and $Q \approx_\ell^{\mathsf{c}} Q_1$. Hence $P_1 \approx_\ell^{\mathsf{c}} Q_1$ and $P_2 \approx_\ell^{\mathsf{p}} Q_2$. We conclude by application Lemma 18. □

Lemma 19. *Let $P, Q \in \mathcal{SAD}$ such that $P \overset{\tau}{\not\to}_{\mathsf{p}}$, $Q \overset{\tau}{\not\to}_{\mathsf{p}}$ and $P \approx_\ell^{\mathsf{c}} Q$. Assume that for all P', Q', $|P'| + |Q'| < |P| + |Q|$ and $P' \approx_\ell^{\mathsf{c}} Q'$ implies $P' \approx_\ell^{\mathsf{p}} Q'$. We have $skel(P) = skel(Q)$.*

Proof. Consider first that for all d, $P \not\xrightarrow{\nu x.out(d,x)}_{\mathsf{p}}$ and $Q \not\xrightarrow{\nu x.out(d,x)}_{\mathsf{p}}$. By applying Lemma 17, we deduce that $skel(P) = skel(Q)$. By applying Corollary 4, we also deduce that $\{out(d) \in skel(P) \mid d \in \mathcal{C}h\} = \{out(d) \in skel(Q) \mid d \in \mathcal{C}h\}$. Therefore, it only remains to prove that $\{in(d) \in skel(P) \mid d \in \mathcal{C}h\} = \{in(d) \in skel(Q) \mid d \in \mathcal{C}h\}$.

Consider $P \xrightarrow{in(d,M)}_{\mathsf{c}} P'$ such that $in(d) \notin skel(Q)$. Since $P \approx_\ell^{\mathsf{c}} Q$, then $Q \xRightarrow{in(d,M)}_{\mathsf{c}} Q'$ for some Q'. As $in(d) \notin skel(Q)$ and $Q \not\xrightarrow{\nu x.out(d,x)}_{\mathsf{p}}$, we deduce that $Q \xrightarrow{\tau}_{\mathsf{c}} Q'' \xRightarrow{in(d,M)}_{\mathsf{c}} Q'$ where $Q \xrightarrow{\tau}_{\mathsf{c}} Q''$ is an internal communication on some public channel c. Since $\{out(d) \in skel(P) \mid d \in \mathcal{C}h\} = \{out(d) \in skel(Q) \mid d \in \mathcal{C}h\}$, we deduce that:

- $P = \nu\tilde{k}.(\mathsf{out}^{\mathsf{ho}}(c,u).R_1 \mid \mathsf{in}^{\mathsf{ho}}(d,x).R_2 \mid R_3)$.
- $Q = \nu\tilde{k}'.(\mathsf{out}^{\mathsf{ho}}(c,v).S_1 \mid \mathsf{in}^{\mathsf{ho}}(c,x).S_2 \mid S_3)$

Moreover, Corollary 4 also tells us that $P_1 = \nu\tilde{k}.(R_1 \mid \mathsf{in}^{\mathsf{ho}}(d,x).R_2 \mid R_3 \mid \{^u/_z\}) \approx_\ell^{\mathsf{p}} \nu\tilde{k}'.(S_1 \mid \mathsf{in}^{\mathsf{ho}}(c,x).S_2 \mid S_3 \mid \{^v/_z\}) = Q_1$. Since $P, Q \in \mathcal{D}(\mathsf{p})$, we can assume w.l.o.g. that $P_1 \overset{\tau}{\not\to}_{\mathsf{p}}$, $Q_1 \overset{\tau}{\not\to}_{\mathsf{p}}$. As $P_1 \approx_\ell^{\mathsf{p}} Q_1$, $skel(P_1) = skel(Q_1)$. Hence $in(d) \in skel(S_1)$ or $in(d) \in skel(S_3)$. As we assumed $in(d) \notin skel(Q)$, $in(d) \in skel(S_1)$. Therefore, $Q = \nu\tilde{k}'.(\mathsf{out}^{\mathsf{ho}}(c,v).(\mathsf{in}^{\mathsf{ho}}(d,y).S_1' \mid S_1'') \mid \mathsf{in}^{\mathsf{ho}}(c,x).S_2 \mid S_3)$.

Note that $Q \xrightarrow{in(c,N)}_{\mathsf{c}} Q_2 = \nu\tilde{k}'.(\mathsf{out}^{\mathsf{ho}}(c,v).(\mathsf{in}^{\mathsf{ho}}(d,y).S_1' \mid S_1'') \mid S_2\{^N/_y\} \mid S_3)$. As $Q \approx_\ell^{\mathsf{c}} P$, there exists $P \xRightarrow{in(c,N)}_{\mathsf{c}} P_2$ and $Q_2 \approx_\ell^{\mathsf{c}} P_2$ for some P_2. Since $|Q_2| + |P_2| < |P| + |Q|$, $Q_2 \approx_\ell^{\mathsf{c}} P_2$ implies $Q_2 \approx_\ell^{\mathsf{p}} P_2$. Note that in Q_2, $\mathsf{out}^{\mathsf{ho}}(c,v)$ and $\mathsf{in}^{\mathsf{ho}}(d,y)$ are sequential. As $Q_2 \approx_\ell^{\mathsf{p}} P_2$, P_2 should also contain a similar sequence of $\mathsf{out}^{\mathsf{ho}}(c,u')$ and $\mathsf{in}^{\mathsf{ho}}(d,y')$ for some u' and y'. Since $P \xRightarrow{in(c,N)}_{\mathsf{c}} P_2$, we deduce that this sequence should appear either in $\mathsf{out}^{\mathsf{ho}}(c,u).R_1$ or in $\mathsf{in}^{\mathsf{ho}}(d,x).R_2$ or in R_3. However, by definition of strong action determinate, there cannot be any instance of $\mathsf{in}^{\mathsf{ho}}(d,y)$ in R_1 or R_3 because of $\mathsf{in}^{\mathsf{ho}}(d,x).R_2$. Similarly, there cannot be any instance of $\mathsf{out}^{\mathsf{ho}}(c,u')$ for some u' in $\mathsf{in}^{\mathsf{ho}}(d,x).R_2$ or R_3 because of $\mathsf{out}^{\mathsf{ho}}(c,u).R_1$. Thus we obtain a contradiction and so $in(d) \in skel(Q)$. □

Lemma 20. *Let $P, Q \in \mathcal{SAD}$ such that $P \overset{\tau}{\not\to}_{\mathsf{p}}$ and $Q \overset{\tau}{\not\to}_{\mathsf{p}}$. If $P \approx_\ell^{\mathsf{c}} Q$ and $skel(P) = skel(Q)$ then $P \approx_\ell^{\mathsf{p}} Q$.*

Proof. Thanks to Lemma 17, we know that the result trivially holds if $\mathit{skel}(P)$ contains only inputs. Thus, let us consider $P = \nu\tilde{k}.(\mathsf{out}^{\mathsf{ho}}(c, u).R_1 \mid R_2)$. Note that by Corollary 4 and since $P \in \mathcal{D}(\mathsf{p})$, we know that $Q = \nu\tilde{k}'.(\mathsf{out}^{\mathsf{ho}}(c, v).S_1 \mid S_2)$ with $\nu\tilde{k}.(R_1 \mid R_2 \mid \{^{u}/_{z}\}) \approx^{\mathsf{p}}_{\ell} \nu\tilde{k}'.(S_1 \mid S_2 \mid \{^{v}/_{z}\})$.

Therefore, let us define the relation $\mathcal{R}$ such that $P' \mathrel{\mathcal{R}} Q'$ iff either $P' \approx^{\mathsf{p}}_{\ell} Q'$ or the following properties hold:

- $\mathit{skel}(P') = \mathit{skel}(Q')$,
- $P' = \nu\tilde{k}.(\mathsf{out}^{\mathsf{ho}}(c, u).R_1 \mid R_2')$, $Q = \nu\tilde{k}'.(\mathsf{out}^{\mathsf{ho}}(c, v).S_1 \mid S_2')$ for some R_2', S_2', and
- $\nu\tilde{k}.(R_1 \mid R_2' \mid \{^{u}/_{z}\}) \approx^{\mathsf{p}}_{\ell} \nu\tilde{k}'.(S_1 \mid S_2' \mid \{^{v}/_{z}\})$.

Note that $P \mathrel{\mathcal{R}} Q$. We show that $\mathcal{R} \subseteq \approx^{\mathsf{p}}_{\ell}$. Consider $P' \mathrel{\mathcal{R}} Q'$. By definition of $\mathcal{R}$, if $P' \approx^{\mathsf{p}}_{\ell} Q'$ then the result trivially holds. Therefore, we consider the other part of the definition of $\mathcal{R}$ and we prove the P', Q' satisfies the definition of $\approx^{\mathsf{p}}_{\ell}$

- We know that $\nu\tilde{k}.(R_1 \mid R_2' \mid \{^{u}/_{z}\}) \approx^{\mathsf{p}}_{\ell} \nu\tilde{k}'.(S_1 \mid S_2' \mid \{^{v}/_{z}\})$. Hence we deduce that $\phi(P') \sim \phi(Q')$.
- Assume $P' \xrightarrow{\nu z.out(c,z)} P''$. Since P' is strongly action determinate, we deduce that $P'' = \nu\tilde{k}.(R_1 \mid R_2' \mid \{^{u}/_{z}\})$. We also have $Q' \xrightarrow{\nu z.out(c,z)} Q'' = \nu\tilde{k}'.(S_1 \mid S_2' \mid \{^{v}/_{z}\})$ and we also have $P'' \approx^{\mathsf{p}}_{\ell} Q''$. Hence $P'' \mathrel{\mathcal{R}} Q''$.
- Assume $P' \xrightarrow{a} P''$ with a different from $\nu z.out(c, z)$. In such a case, $P'' = \nu\tilde{k}.(\mathsf{out}^{\mathsf{ho}}(c, u).R_1 \mid R_2'')$ for some R_2''. Note that $P' \xrightarrow{\nu z.out(c,z).a}_{\mathsf{p}} P''' = \nu\tilde{k}.(R_1 \mid R_2'' \mid \{^{u}/_{z}\})$. Since $\nu\tilde{k}.(R_1 \mid R_2' \mid \{^{u}/_{z}\}) \approx^{\mathsf{p}}_{\ell} \nu\tilde{k}'.(S_1 \mid S_2' \mid \{^{v}/_{z}\})$ and $\mathit{skel}(P') = \mathit{skel}(Q')$, we deduce that $Q' \xRightarrow{\nu z.out(c,z).a}_{\mathsf{p}} Q''' = \nu\tilde{k}'.(S_1 \mid S_2'' \mid \{^{v}/_{z}\})$ for some S_2'' and $P''' \approx^{\mathsf{p}}_{\ell} Q'''$. However, note that we also have $Q' \xRightarrow{a.\nu z.out(c,z)}_{\mathsf{p}} Q'''$ with $Q' \overset{a}{\Rightarrow}_{\mathsf{p}} \nu\tilde{k}'.(\mathsf{out}^{\mathsf{ho}}(c, v).S_1 \mid S_2'') = Q''$. As $\mathit{skel}(P') = \mathit{skel}(Q')$ and $P''' \approx^{\mathsf{p}}_{\ell} Q'''$, we deduce that $\mathit{skel}(S_2'') = \mathit{skel}(R_2'')$ and so $\mathit{skel}(P'') = \mathit{skel}(Q'')$. We conclude that $P'' \mathrel{\mathcal{R}} Q''$. □

Theorem 10. *When restricted to $\mathcal{SAD}$, we have $\approx^{\mathsf{c}}_{\ell} \subseteq \approx^{\mathsf{p}}_{\ell}$.*

Proof. Consider $P, Q \in \mathcal{SAD}$ such that $P \approx^{\mathsf{c}}_{\ell} Q$. We prove this property by induction on $|P| + |Q|$. The base case ($|P| + |Q| = 0$) being trivial, we focus on the inductive step.

Assume $P \xrightarrow{\tau}_{\mathsf{p}} P'$. Since $P \in \mathcal{D}(\mathsf{p})$, we know that $P \approx^{\mathsf{p}}_{\ell} P'$. By Lemma 13, $P \approx^{\mathsf{p}}_{\ell} P'$ implies $P \approx^{\mathsf{c}}_{\ell} P'$. Hence $P' \approx^{\mathsf{c}}_{\ell} Q$. Since $|P'| + |Q| < |P| + |Q|$, we can apply our inductive hypothesis which gives us $P' \approx^{\mathsf{p}}_{\ell} Q$. As $P \approx^{\mathsf{p}}_{\ell} P'$, we conclude $P \approx^{\mathsf{p}}_{\ell} Q$. By symmetry, the same proof holds when $Q \xrightarrow{\tau}_{\mathsf{p}} Q'$.

Therefore, assume that $P \overset{\tau}{\not\rightarrow}_{\mathsf{p}}$ and $Q \overset{\tau}{\not\rightarrow}_{\mathsf{p}}$. Thanks to Lemma 19, we deduce that $\mathit{skel}(P) = \mathit{skel}(Q)$. We conclude by applying Lemma 20. □

Appendix K. Proof of Theorem 11

Theorem 11. *When restricted to I/O-unambiguous processes, we have that $\approx^{\mathsf{p}}_{r} = \approx^{\mathsf{e}}_{r}$ but $\approx^{\mathsf{e}}_{r} \subsetneq \approx^{\mathsf{c}}_{r}$ for $r \in \{\ell, t\}$.*

Proof. From Theorems 5, 6 and 4, we already know that $\approx^{\mathsf{e}}_{r} \subseteq \approx^{\mathsf{p}}_{r} \cap \approx^{\mathsf{e}}_{r}$ for $r \in \{\mathit{lbl}, m, t\}$. Hence, for $r \in \{\mathit{lbl}, m, t\}$, we only need to prove that $\approx^{\mathsf{p}}_{r} \subseteq \approx^{\mathsf{e}}_{r}$ and $\approx^{\mathsf{e}}_{r} \subseteq \approx^{\mathsf{c}}_{r}$ to obtain the result.

Proof of $\approx_t^{\mathsf{p}} \subseteq \approx_t^{\mathsf{e}}$: Let A and B to honest I/O-unambiguous processes such that $A \approx_t^{\mathsf{p}} B$. Let $A \overset{\mathsf{tr}}{\Rightarrow}_{\mathsf{e}} A'$. By definition, we know that there exist $\ell_1, \dots, \ell_n$ and extended processes $A_0, \dots, A_n$ such that:

- tr is $\ell_1 \dots \ell_n$ where the τ are removed
- $A_0 = A$, $A_n = A'$
- $A_0 \xrightarrow{\ell_1}_{\mathsf{e}} A_1 \xrightarrow{\ell_2}_{\mathsf{e}} \dots \xrightarrow{\ell_n}_{\mathsf{e}} A_n$.

Note that since A is honest, the rules C-Env, C-Open, C-Eav, C-OEav are never applied in the derivation. The idea is to

$\approx_r^{s_1} = \approx_r^{s_2}$ for $r \in \{\ell, o, m, t\}$ and $s_1, s_2 \in \{\mathsf{c}, \mathsf{p}, \mathsf{e}\}$

We first focus on the proof of $\approx_r^{s_1} = \approx_r^{s_2}$ for $r \in \{\ell, o, m, t\}$ and $s_1, s_2 \in \{\mathsf{c}, \mathsf{p}, \mathsf{e}\}$. □

References

[1] M. Abadi and C. Fournet, Mobile values, new names, and secure communication, in: *28th Symposium on Principles of Programming Languages (POPL'01)*, H.R. Nielson, ed., ACM, London, UK, 2001, pp. 104–115.

[2] M. Abadi and C. Fournet, Private authentication, *Theor. Comput. Sci.* **322**(3) (2004), 427–476. doi:10.1016/j.tcs.2003.12.023.

[3] M. Abadi and A.D. Gordon, A calculus for cryptographic protocols: The spi calculus, *Inf. Comput.* **148**(1) (1999), 1–70. doi:10.1006/inco.1998.2740.

[4] B. Adida, Helios: Web-based open-audit voting, in: *17th Conference on Security Symposium (SS'08)*, USENIX Association, 2008, pp. 335–348, http://dl.acm.org/citation.cfm?id=1496711.1496734.

[5] M. Arapinis, V. Cheval and S. Delaune, Verifying privacy-type properties in a modular way, in: *Proceedings of the 25th IEEE Computer Security Foundations Symposium (CSF'12)*, V. Cortier and S. Zdancewic, eds, IEEE Computer Society Press, Cambridge Massachusetts, USA, 2012, pp. 95–109.

[6] M. Arapinis, T. Chothia, E. Ritter and M. Ryan, Analysing unlinkability and anonymity using the applied pi calculus, in: *Proc. 23rd Computer Security Foundations Symposium (CSF'10)*, IEEE Computer Society Press, 2010, pp. 107–121.

[7] M. Arapinis, L. Mancini, E. Ritter, M. Ryan, N. Golde, K. Redon and R. Borgaonkar, New privacy issues in mobile telephony: Fix and verification, in: *19th Conference on Computer and Communications Security (CCS'12)*, ACM Press, 2012, pp. 205–216.

[8] A. Armando, D.A. Basin, Y. Boichut, Y. Chevalier, L. Compagna, J. Cuéllar, P.H. Drielsma, P.-C. Héam, O. Kouchnarenko, J. Mantovani, S. Mödersheim, D. von Oheimb, M. Rusinowitch, J. Santiago, M. Turuani, L. Viganò and L. Vigneron, The AVISPA tool for the automated validation of Internet security protocols and applications, in: *Proc. 17th International Conference on Computer Aided Verification (CAV'05)*, Lecture Notes in Computer Science, Springer, 2005, pp. 281–285.

[9] K. Babel, V. Cheval and S. Kremer, On communication models when verifying equivalence properties, in: *6th International Conference on Principles of Security and Trust (POST'17)*, Lecture Notes in Computer Science, Vol. 10204, Springer, Uppsala, Sweden, 2017, pp. 141–163. doi:10.1007/978-3-662-54455-6.

[10] D. Baelde, S. Delaune and L. Hirschi, Partial order reduction for security protocols, in: *Proceedings of the 26th International Conference on Concurrency Theory (CONCUR'15)*, L. Aceto and D. de Frutos-Escrig, eds, Leibniz International Proceedings in Informatics, Vol. 42, Leibniz-Zentrum für Informatik, Madrid, Spain, 2015, pp. 497–510, http://www.lsv.ens-cachan.fr/Publis/PAPERS/PDF/BDH-concur15.pdf. doi:10.4230/LIPIcs.CONCUR.2015.497.

[11] B. Blanchet, Automatic verification of correspondences for security protocols, *Journal of Computer Security* **17**(4) (2009), 363–434. doi:10.3233/JCS-2009-0339.

[12] B. Blanchet, M. Abadi and C. Fournet, Automated verification of selected equivalences for security protocols, *Journal of Logic and Algebraic Programming* **75**(1) (2008), 3–51. doi:10.1016/j.jlap.2007.06.002.

[13] R. Chadha, V. Cheval, Ş. Ciobâcă and S. Kremer, Automated verification of equivalence properties of cryptographic protocol, *ACM Transactions on Computational Logic* **17**(4) (2016), 1–32. doi:10.1145/2926715.

[14] V. Cheval, H. Comon-Lundh and S. Delaune, Trace equivalence decision: Negative tests and non-determinism, in: *Proc. 18th ACM Conference on Computer and Communications Security (CCS'11)*, ACM, 2011.

[15] V. Cheval, V. Cortier and S. Delaune, Deciding equivalence-based properties using constraint solving, *Theoretical Computer Science* **492** (2013), 1–39. doi:10.1016/j.tcs.2013.04.016.

[16] V. Cheval, S. Kremer and I. Rakotonirina, DEEPSEC: Deciding equivalence properties in security protocols – theory and practice, in: *Proceedings of the 39th IEEE Symposium on Security and Privacy (S&P'18)*, IEEE Computer Society Press, San Francisco, CA, USA, 2018, pp. 525–542. doi:10.1109/SP.2018.00033.

[17] V. Cheval, S. Kremer and I. Rakotonirina, DeepSec 1.1, 2019, https://github.com/DeepSec-prover/deepsec/tree/ae7a64e9023df242370b011dfa82a7586ac7a772.

[18] V. Cortier, S. Delaune and A. Dallon, SAT-Equiv: An efficient tool for equivalence properties, in: *Proceedings of the 30th IEEE Computer Security Foundations Symposium (CSF'17)*, IEEE Computer Society Press, 2017, pp. 481–494. doi:10.1109/CSF.2017.15.

[19] V. Cortier, D. Galindo and M. Turuani, A formal analysis of the Neuchâtel e-voting protocol, in: *IEEE European Symposium on Security and Privacy (EuroS&P)*, 2018.

[20] C.J.F. Cremers, The Scyther Tool: Verification, falsification, and analysis of security protocols, in: *Proc. 20th International Conference on Computer Aided Verification (CAV'08)*, Lecture Notes in Computer Science, Vol. 5123, Springer, 2008, pp. 414–418. doi:10.1007/978-3-540-70545-1_38.

[21] S. Delaune, S. Kremer and M.D. Ryan, Verifying privacy-type properties of electronic voting protocols, *Journal of Computer Security* **17**(4) (2009), 435–487. doi:10.3233/JCS-2009-0340.

[22] N. Dong, H. Jonker and J. Pang, Analysis of a receipt-free auction protocol in the applied pi calculus, in: *Proc. International Workshop on Formal Aspects in Security and Trust (FAST'10)*, S. Etalle and J. Guttman, eds, Pisa, Italy, 2010, To appear.

[23] P.T. Force, PKI for machine readable travel documents offering ICC read-only access, Technical Report, International Civil Aviation Organization, 2004.

[24] J.K. Millen and V. Shmatikov, Constraint solving for bounded-process cryptographic protocol analysis, in: *Proc. 8th Conference on Computer and Communications Security*, ACM Press, 2001, pp. 166–175.

[25] L.C. Paulson, The inductive approach to verifying cryptographic protocols, *Journal of Computer Security* **6**(1/2) (1998), 85–128. doi:10.3233/JCS-1998-61-205.

[26] P.Y.A. Ryan, S.A. Schneider, M. Goldsmith, G. Lowe and A.W. Roscoe, *Modelling and Analysis of Security Protocols*, Addison Wesley, 2000.

[27] B. Schmidt, S. Meier, C. Cremers and D. Basin, The TAMARIN prover for the symbolic analysis of security protocols, in: *Proc. 25th International Conference on Computer Aided Verification (CAV'13)*, Lecture Notes in Computer Science, Vol. 8044, Springer, 2013, pp. 696–701.

[28] F.J. Thayer Fabrega, J.C. Herzog and J.D. Guttman, Strand spaces: Proving security protocols correct, *Journal of Computer Security* **7**(2/3) (1999), 191–230. doi:10.3233/JCS-1999-72-304.

[29] A. Tiu and J.E. Dawson, Automating open bisimulation checking for the spi calculus, in: *Proc. 23rd Computer Security Foundations Symp. (CSF'10)*, IEEE Comp. Soc., 2010, pp. 307–321.