Psi-Calculi Revisited: Connectivity and Compositionality

Johannes Åman Pohjola[0000−0002−6406−7875]

Data61/CSIRO, Sydney, Australia
University of New South Wales, Sydney, Australia
johannes.amanpohjola@data61.csiro.au

Abstract. Psi-calculi is a parametric framework for process calculi similar to popular pi-calculus extensions such as the explicit fusion calculus, the applied pi-calculus and the spi calculus. Mechanised proofs of standard algebraic and congruence properties of bisimilarity apply to all calculi within the framework.
A limitation of psi-calculi is that communication channels must be symmetric and transitive. Here, we give a new operational semantics to psi-calculi that allows us to lift these restrictions and simplify some of the proofs. The key technical innovation is to annotate transitions with a *provenance*—a description of the scope and channel they originate from. We give mechanised proofs that our extension is conservative, and that the standard algebraic and congruence properties of bisimilarity are maintained. We show correspondence with a reduction semantics and barbed bisimulation. We show how a pi-calculus with preorders that was previously beyond the scope of psi-calculi can be captured, and how to encode mixed choice under very strong quality criteria.

Keywords: Process algebra · Psi-calculi · Nominal logic · Interactive theorem proving · Bisimulation

1 Introduction

This report is mainly concerned with *channel connectivity*, by which we mean the relationship that describes which input channels are connected to which output channels in a setting with message-passing concurrency. In the pi-calculus [18], channel connectivity is syntactic identity: in the process

$$\underline{a}(x).P \mid \overline{b}\,y.Q$$

where one parallel component is waiting to receive on channel a and the other is waiting to send on channel b, interaction is possible only if $a = b$.

Variants of the pi-calculus may have more interesting channel connnectivity. The explicit fusion calculus pi-F [9] extends the pi-calculus with a primitive for *fusing* names; once fused, they are treated as being for all purposes one and the same. Channel connectivity is then given by the equivalence closure of the name fusions. For example, if we extend the above example with the fusion $(a = b)$

$$\underline{a}(x).P \mid \overline{b}\,y.Q \mid (a = b)$$

then communication is possible. Other examples may be found in e.g. calculi for wireless communication [19], where channel connectivity can be used to directly model the network's topology.

Psi-calculi [1] is a family of applied process calculi, where standard meta-theoretical results, such as the algebraic laws and congruence properties of bisimulation, have been established once and for all through mechanised proofs [2] for all members of the family. Psi-calculi generalises e.g. the pi-calculus and the explicit fusion calculus in several ways. In place of atomic names it allows channels and messages to be taken from an (almost) freely chosen term language. In place of fusions, it admits the formulas of an (almost) freely chosen logic as first-class processes. Channel connectivity is determined by judgements of said logic, with one restriction: the connectivity thus induced must be symmetric and transitive.

The main contribution here is a new way to define the semantics of psi-calculi that lets us lift this restriction, without sacrificing any of the algebraic laws and compositionality properties. It is worth noting that this was previously believed to be impossible: Bengtson et al. [1, p. 14] even offer counterexamples to the effect that without symmetry and transitivity, scope extension is unsound. However, a close reading reveals that these counterexamples apply only to their particular choice of labelled semantics, and do not rule out the possibility that the counterexamples could be invalidated by a rephrasing of the labelled semantics such as ours.

The price we pay for this increased generality is more complicated transition labels: we decorate input and output labels with a *provenance* that keeps track of which prefix a transition originates from. The idea is that if I am an input label and you are an output label, we can communicate if my subject is your provenance, and vice versa. This is offset by other simplifications of the semantics and associated proofs that provenances enable.

Contributions This report makes the following specific technical contributions:

- We define a new semantics of psi-calculi that lifts the requirement that channel connectivity must be symmetric and transitive, using the novel technical device of provenances. (Section 2)
- We prove that strong bisimulation is a congruence and satisfies the usual algebraic laws such as scope extension. Interestingly, provenances can be ignored for the purpose of bisimulation. These proofs are machine-checked[1] in Nominal Isabelle [24]. (Section 3.1)
- We prove, again using Nominal Isabelle, that this paper's developments constitute a conservative extension of the original psi-calculi. (Section 3.2)
- We further validate our semantics by defining a reduction semantics and strong barbed congruence, and showing that they agree with their labelled counterparts. (Section 3.2)

[1] Isabelle proofs are available at `https://github.com/IlmariReissumies/newpsi`

- We capture a pi-calculus with preorders by Hirschkoff et al. [11], that was previously beyond the scope of psi-calculi because of its non-transitive channel connectivity. The bisimilarity we obtain turns out to coincide with that of Hirschkoff et al. (Section 4.1)
- We exploit non-transitive connectivity to show that mixed choice is a derived operator of psi-calculi in a very strong sense: its encoding is fully abstract and satisfies strong operational correspondence. (Section 4.2)

This report constitutes supplementary material for the author's paper of the same title, to appear at the 14th International Federated Conference on Distributed Computing Techniques (FORTE 2019). The final publication is available at Springer via `http://dx.doi.org`.

In particular, this report includes an appendix, which gives justification for all formal claims made: pointers to the relevant parts of the Isabelle formalisation for theorems with machine-checked proofs, and full proof details for hand-written proofs.

2 Definitions

This section introduces core definitions such as syntax and semantics. Many definitions are shared with the original presentation of psi-calculi, so this section also functions as a recapitulation of [1]. We will highlight the places where the two differ.

We assume a countable set of *names* $\mathcal{N}$ ranged over by $a, b, c, \ldots, x, y, z$. A *nominal set* [8] is a set equipped with a permutation action $\cdot$; intuitively, if $X \in \mathbf{X}$ and $\mathbf{X}$ is a nominal set, then $(x\ y) \cdot X$, which denotes X with all occurences of the name x swapped for y and vice versa, is also an element of $\mathbf{X}$. $\mathsf{n}(X)$ (the *support* of X) is, intuitively, the set of names such that swapping them changes X. We write $a\#X$ ("a is fresh in X) for $a \notin \mathsf{n}(X)$. A nominal set $\mathbf{X}$ has *finite support* if for every $X \in \mathbf{X}$, $\mathsf{n}(X)$ is finite. A function symbol f is *equivariant* if $p \cdot f(x) = f(p \cdot x)$; this generalises to n-ary function symbols in the obvious way. Whenever we define inductive syntax with names, it is implicitly quotiented by permutation of bound names, so e.g. $(\nu x)\overline{a}\langle x\rangle = (\nu y)\overline{a}\langle y\rangle$ if $x, y\#a$.

Psi-calculi is parameterised on an arbitrary term language and a logic of environmental assertions:

Definition 1 (Parameters). *A* psi-calculus *is a 7-tuple* $(\mathbf{T}, \mathbf{A}, \mathbf{C}, \vdash, \otimes, \mathbf{1}, \dot{\rightarrow})$ *with three finitely supported nominal sets:*

1. $\mathbf{T}$*, the* terms*, ranged over by* M, N, K, L, T*;*
2. $\mathbf{A}$*, the* assertions*, ranged over by* Ψ*; and*
3. $\mathbf{C}$*, the* conditions*, ranged over by* φ*.*

We assume each of the above is equipped with a substitution function $[\_ := \_]$ *that substitutes (sequences of) terms for names. The remaining three parameters are equivariant function symbols written in infix:*

$\vdash : \mathbf{A} \times \mathbf{C} \Rightarrow \mathbf{bool}$ *(entailment)* $\quad \otimes : \mathbf{A} \times \mathbf{A} \Rightarrow \mathbf{A}$ *(composition)*
$\mathbf{1} : \mathbf{A}$ *(unit)* $\quad \dot{\rightarrow} : \mathbf{T} \times \mathbf{T} \Rightarrow \mathbf{C}$ *(channel connectivity)*

Intuitively, $M \dot{\rightarrow} K$ means the prefix M can send a message to the prefix K. The substitution functions must satisfy certain natural criteria wrt. their treatment of names; see [1] for the details.

Definition 2 (Static equivalence). *Two assertions Ψ, Ψ' are* statically equivalent*, written $\Psi \simeq \Psi'$, if $\forall \varphi.\, \Psi \vdash \varphi \;\Leftrightarrow\; \Psi' \vdash \varphi$.*

Definition 3 (Valid parameters). *A psi-calculus is* valid *if $(\mathbf{A}/\simeq, \otimes, \mathbf{1})$ form an abelian monoid.*

Note that since the abelian monoid is closed, static equivalence is preserved by composition. Henceforth we will only consider valid psi-calculi. The original presentation of psi-calculi had $\dot{\leftrightarrow}$ for channel equivalence in place of our $\dot{\rightarrow}$, and required that channel equivalence be symmetric (formally, $\Psi \vdash M \dot{\leftrightarrow} K$ iff $\Psi \vdash K \dot{\leftrightarrow} M$) and transitive.

Definition 4 (Process syntax). *The* processes *(or* agents*)* $\mathbf{P}$*, ranged over by P, Q, R, are inductively defined by the grammar*

$P :=\ \mathbf{0}$	*(nil)*	$(\!\lvert \Psi \rvert\!)$	*(assertion)*
$\quad \overline{M}\, N.P$	*(output)*	$\quad \underline{M}(\lambda \widetilde{x})N.P$	*(input)*
$\quad \mathbf{case}\ \widetilde{\varphi} : \widetilde{P}$	*(case)*	$\quad P \mid Q$	*(parallel composition)*
$\quad (\nu x)P$	*(restriction)*	$\quad !P$	*(replication)*

A process is assertion guarded *(*guarded *for short) if all assertions occur underneath an input or output prefix. We require that in $!P$, P is guarded; that in $\mathbf{case}\ \widetilde{\varphi} : \widetilde{P}$, all $\widetilde{P}$ are guarded; and that in $\underline{M}(\lambda \widetilde{x})N\,.\,P$ it holds that $\widetilde{x} \subseteq \mathsf{n}(N)$. We will use P_G, Q_G to range over guarded processes.*

Restriction, replication and parallel composition are standard. $\overline{M}\, N.P$ is a process ready to send the message N on channel M, and then continue as P. Similarly, $\underline{M}(\lambda \widetilde{x})N.P$ is a process ready to receive a message on channel M that matches the pattern $(\lambda \widetilde{x})N$. The process $(\!\lvert \Psi \rvert\!)$ asserts a fact Ψ about the environment. Intuitively, $(\!\lvert \Psi \rvert\!) \mid P$ means that P executes in an environment where all conditions entailed by Ψ hold. P may itself contain assertions that add or retract conditions. Environments can evolve dynamically: as a process reduces, assertions may become unguarded and thus added to the environment. $\mathbf{case}\ \widetilde{\varphi} : \widetilde{P}$ is a process that may act as any P_i whose guard φ_i is entailed by the environment. For discussion of why replication and case must be guarded we refer to [1,15].

The assertion environment of a process is described by its *frame*:

Definition 5 (Frames). *The* frame *of P, written $\mathcal{F}(P) = (\nu \widetilde{b_P})\Psi_P$ where $\widetilde{b}_P$ bind into Ψ_P, is defined as*

$$\mathcal{F}(\!(\Psi)\!) = (\nu \epsilon)\Psi \qquad \mathcal{F}(P \mid Q) = \mathcal{F}(P) \otimes \mathcal{F}(Q) \qquad \mathcal{F}((\nu x)P) = (\nu x)\mathcal{F}(P)$$

$$\mathcal{F}(P) = \mathbf{1} \ \textit{otherwise}$$

where name-binding and composition of frames is defined as $(\nu x)(\nu\widetilde{b}_P)\Psi_P = (\nu x, \widetilde{b}_P)\Psi_P$*, and, if* $\widetilde{b}_P\#\widetilde{b}_Q, \Psi_Q$ *and* $\widetilde{b}_Q\#\Psi_P$*,*

$$(\nu\widetilde{b}_P)\Psi_P \otimes (\nu\widetilde{b}_Q)\Psi_Q = (\nu\widetilde{b}_P, \widetilde{b}_Q)\Psi_P \otimes \Psi_Q.$$

We extend entailment to frames as follows: $\mathcal{F}(P) \vdash \varphi$ holds if, for some $\widetilde{b}_P, \Psi_P$ such that $\mathcal{F}(P) = (\nu\widetilde{b}_P)\Psi_P$ and $\widetilde{b}_P\#\varphi$, $\Psi_P \vdash \varphi$. The freshness side-condition $\widetilde{b}_P\#\varphi$ is important because it allows assertions to be used for representing local state. By default, the assertion environment is effectively a form of global non-monotonic state, which is not always appropriate for modelling distributed processes. With ν-binding we recover locality by writing e.g. $(\nu x)((\!|x = M|\!) \mid P)$ for a process P with a local variable x.

The notion of *provenance* is the main novelty of our semantics. It is the key technical device used to make our semantics compositional:

Definition 6 (Provenances). *The* provenances Π*, ranged over by* π*, are either* $\bot$ *or of form* $(\nu\widetilde{x};\widetilde{y})M$*, where* M *is a term, and* $\widetilde{x}, \widetilde{y}$ *bind into* M*.*

We write M for $(\nu\epsilon;\epsilon)M$. When $\widetilde{x}, \widetilde{y}\#\widetilde{x'}, \widetilde{y'}$ and $\widetilde{x}\#\widetilde{y}$, we interpret the expression $(\nu\widetilde{x};\widetilde{y})(\nu\widetilde{x'};\widetilde{y'})M$ as $(\nu\widetilde{x}\,\widetilde{x'};\widetilde{y}\,\widetilde{y'})M$. Furthermore, we identify $(\nu\widetilde{x};\widetilde{y})\bot$ and $\bot$. Let $\pi\downarrow$ denote the result of moving all binders from the outermost binding sequence to the innermost; that is, $(\nu\widetilde{x};\widetilde{y})M\downarrow = (\nu\epsilon;\widetilde{x},\widetilde{y})M$. Similarly, $\pi\downarrow\widetilde{z}$ denotes the result of inserting $\widetilde{z}$ at the end of the outermost binding sequence: formally, $(\nu\widetilde{x};\widetilde{y})M\downarrow\widetilde{z} = (\nu\widetilde{x},\widetilde{z};\widetilde{y})M$.

Intuitively, a provenance describes the origin of an input or output transition. For example, if an output transition is annotated with $(\nu\widetilde{x};\widetilde{y})M$, the sender is an output prefix with subject M that occurs underneath the ν-binders $\widetilde{x}, \widetilde{y}$. For technical reasons, these binders are partitioned into two distinct sequences. The intention is that $\widetilde{x}$ are the frame binders, while $\widetilde{y}$ contains binders that occur underneath case and replication; these are not part of the frame, but may nonetheless bind into M. We prefer to keep them separate because the $\widetilde{x}$ binders are used for deriving $\vdash$ judgements, but $\widetilde{y}$ are not (cf. Definition 5).

Definition 7 (Labels). *The* labels $\mathbf{L}$*, ranged over by* α, β*, are:*

$$\overline{M}\,(\nu\widetilde{x})N \text{ (output)} \qquad \underline{M}\,N \text{ (input)} \qquad \tau \text{ (silent)}$$

The bound names of α*, written* $\mathrm{bn}(alpha)$*, is* $\widetilde{x}$ *if* $\alpha = \overline{M}\,(\nu\widetilde{x})N$ *and* ϵ *otherwise. The subject of* α*, written* $\mathrm{subj}(\alpha)$*, is* M *if* $\alpha = \overline{M}(\nu\widetilde{x})N$ *or* $\alpha = \underline{M}N$*. Analogously, the object of* α*, written* $\mathrm{obj}(\alpha)$*, is* N *if* $\alpha = \overline{M}\,(\nu\widetilde{x})N$ *or* $\alpha = \underline{M}\,N$*.*

While the provenance describes the origin of a transition, a label describes how it can interact. For example, a transition labelled with $\underline{M}\,N$ indicates readiness to receive a message N from an output prefix with subject M.

Definition 8 (Operational semantics). *The transition relation* $\longrightarrow \subseteq \mathbf{A}\times\mathbf{P}\times\mathbf{L}\times\Pi\times\mathbf{P}$ *is inductively defined by the rules in Table 1. We write* $\Psi \rhd P \xrightarrow[\pi]{\alpha} P'$ *for* $(\Psi, P, \alpha, \pi, P') \in \longrightarrow$*. In transitions,* $\mathrm{bn}(\alpha)$ *binds into* $\mathrm{obj}(\alpha)$ *and* P'*.*

$$\textsc{In}\ \frac{\Psi \vdash K \doteq M}{\Psi \rhd \underline{M}(\lambda\widetilde{y})N \,.\, P \xrightarrow[M]{\underline{K}\ N[\widetilde{y}:=\widetilde{L}]} P[\widetilde{y} := \widetilde{L}]} \qquad \textsc{Out}\ \frac{\Psi \vdash M \doteq K}{\Psi \rhd \overline{M}\, N \,.\, P \xrightarrow[M]{\overline{K} N} P}$$

$$\textsc{ParL}\ \frac{\Psi_Q \otimes \Psi \rhd P \xrightarrow[\pi]{\alpha} P'}{\Psi \rhd P \mid Q \xrightarrow[\pi\downarrow\widetilde{b}_Q]{\alpha} P' \mid Q}\ \mathrm{bn}(\alpha)\#Q$$

$$\textsc{ParR}\ \frac{\Psi_P \otimes \Psi \rhd Q \xrightarrow[\pi]{\alpha} Q'}{\Psi \rhd P \mid Q \xrightarrow[(\nu\widetilde{b}_P)\pi]{\alpha} P \mid Q'}\ \mathrm{bn}(\alpha)\#P$$

$$\textsc{Com}\ \frac{\Psi_Q \otimes \Psi \rhd P \xrightarrow[(\nu\widetilde{b}_P;\widetilde{x})K]{\overline{M}(\nu\widetilde{a})N} P' \qquad \Psi_P \otimes \Psi \rhd Q \xrightarrow[(\nu\widetilde{b}_Q;\widetilde{y})M]{\underline{K}\ N} Q'}{\Psi \rhd P \mid Q \xrightarrow[\bot]{\tau} (\nu\widetilde{a})(P' \mid Q')}\ \widetilde{a}\#Q$$

$$\textsc{Case}\ \frac{\Psi \rhd P_i \xrightarrow[\pi]{\alpha} P' \qquad \Psi \vdash \varphi_i}{\Psi \rhd \mathbf{case}\ \widetilde{\varphi} : \widetilde{P} \xrightarrow[\pi\downarrow]{\alpha} P'} \qquad \textsc{Scope}\ \frac{\Psi \rhd P \xrightarrow[\pi]{\alpha} P'}{\Psi \rhd (\nu b)P \xrightarrow[(\nu b)\pi]{\alpha} (\nu b)P'}\ b\#\alpha, \Psi$$

$$\textsc{Open}\ \frac{\Psi \rhd P \xrightarrow[\pi]{\overline{M}(\nu\widetilde{a})N} P'}{\Psi \rhd (\nu b)P \xrightarrow[(\nu b)\pi]{\overline{M}(\nu\widetilde{a}\cup\{b\})N} P'}\ \begin{array}{l} b\#\widetilde{a}, \Psi, M \\ b \in \mathrm{n}(N) \end{array} \qquad \textsc{Rep}\ \frac{\Psi \rhd P \mid !P \xrightarrow[\pi]{\alpha} P'}{\Psi \rhd !P \xrightarrow[\pi\downarrow]{\alpha} P'}$$

Table 1. Structured operational semantics. A symmetric version of Com is elided. In the rule Com we assume that $\mathcal{F}(P) = (\nu\widetilde{b}_P)\Psi_P$ and $\mathcal{F}(Q) = (\nu\widetilde{b}_Q)\Psi_Q$ where $\widetilde{b}_P$ is fresh for Ψ and Q, $\widetilde{x}$ is fresh for Ψ, Ψ_Q, P, and $\widetilde{b}_Q, \widetilde{y}$ are similarly fresh. In rule ParL we assume that $\mathcal{F}(Q) = (\nu\widetilde{b}_Q)\Psi_Q$ where $\widetilde{b}_Q$ is fresh for Ψ, P, π and α. ParR has the same freshness conditions but with the roles of P, Q swapped. In Open the expression $\tilde{a} \cup \{b\}$ means the sequence $\tilde{a}$ with b inserted anywhere.

The operational semantics differs from [1] mainly by the inclusion of provenances: anything underneath the transition arrows is novel.

The Out rule states that in an environment where M is connected to K, the prefix $\overline{M}\, N$ may send a message N from M to K. The In rule is dual to Out, but also features pattern-matching. If the message is an instance of the pattern, as witnessed by a substitution, that subtitution is applied to the continuation P.

In the Com rule, we see how provenances are used to determine when two processes can interact. Specifically, a communication between P and Q can be derived if P can send a message to M using prefix K, and if Q can receive a message from K using prefix M. Because names occuring in M and K may be local to P and Q respectively, we must be careful not to conflate the local names of one with the other; this is why the provenance records all binding names

that occur above M, K in the process syntax. Note that even though we identify frames and provenances up-to alpha, the Com rule insists that we consider alpha-variants such that the frame binders and the outermost provenance binders coincide. This ensures that the K on Q's label really is the same as the K in the provenance.

It is instructive to compare our Com rule with the original:

$$\textsc{Com-Old}\ \frac{\Psi_Q \otimes \Psi \rhd P \xrightarrow{\overline{M}\,(\nu\widetilde{a})N} P' \qquad \Psi_P \otimes \Psi \rhd Q \xrightarrow{K\,N} Q' \qquad \Psi \otimes \Psi_P \otimes \Psi_Q \vdash M \leftrightarrow K}{\Psi \rhd P \mid Q \xrightarrow{\tau} (\nu\widetilde{a})(P' \mid Q')}\ \widetilde{a}\#Q$$

where $\mathcal{F}(P) = (\nu\widetilde{b}_P)\Psi_P$ and $\mathcal{F}(Q) = (\nu\widetilde{b}_Q)\Psi_Q$ and $\widetilde{b}_P\#\Psi, \widetilde{b}_Q, Q, M, P$ and $\widetilde{b}_Q\#\Psi, \widetilde{b}_Q, Q, K, P$. Here we have no way of knowing if M and K are able to synchronise other than making a channel equivalence judgement. Hence any derivation involving Com-Old makes three channel equivalence judgements: once each in In, Out and Com-Old. With Com we only make one — or more accurately, we make the exact same judgement twice, in In resp. Out. Eliminating the redundant judgements is crucial: the reason Com-Old needs associativity and commutativity is to stitch these three judgements together, particularly when one end of a communication is swapped for a bisimilar process that allows the same interaction via different prefixes.

Note also that Com has fewer freshness side-conditions. A particularly unintuitive aspect of Com-Old is that it requires $\widetilde{b}_P\#M$ and $\widetilde{b}_Q\#K$, but not $\widetilde{b}_P\#K$ and $\widetilde{b}_Q\#M$: we would expect that all bound names can be chosen to be distinct from all free names, but adding the missing freshness conditions makes scope extension unsound [14, pp. 56-57]. With Com, it becomes clear why: because $\widetilde{b}_Q$ binds into M.

All the other rules can fire independently of what the provenance of the premise is. They manipulate the provenance, but only for bookkeeping purposes: in order for the Com rule to be sound, we maintain the invariant that if $\Psi \rhd P \xrightarrow[\pi]{\alpha} P'$, the outer binders of π are precisely the binders of $\mathcal{F}(P)$. Otherwise, the rules are exactly the same as in the original psi-calculi.

The reader may notice a curious asymmetry between the treatment of provenance binders in the ParL and ParR rules. This is to ensure that the order of the provenance binders coincides with the order of the frame binders, and in the frame $\mathcal{F}(P \mid Q)$, the binders of P occur syntactically outside the binders of Q (cf. Definition 5).

3 Meta-theory

In this section, we will derive the standard algebraic and congruence laws of strong bisimulation, develop an alternative formulation of strong bisimulation in terms of a reduction relation and barbed congruence, and show that our extension

of psi-calculi is conservative. While weak equivalences are beyond the scope of the present paper, we believe it is possible (if tedious) to adapt the results about weak bisimilarity from [15] to our setting.

3.1 Bisimulation

We write $\Psi \rhd P \xrightarrow{\alpha} P'$ as shorthand for $\exists \pi.\, \Psi \rhd P \xrightarrow[\pi]{\alpha} P'$. Bisimulation is then defined exactly as in the original psi-calculi:

Definition 9 (Strong bisimulation). *A symmetric relation $\mathcal{R} \subseteq \mathbf{A} \times \mathbf{P} \times \mathbf{P}$ is a* strong bisimulation *iff for every $(\Psi, P, Q) \in \mathcal{R}$*

1. *$\Psi \otimes \mathcal{F}(P) \simeq \Psi \otimes \mathcal{F}(Q)$ (static equivalence)*
2. *$\forall \Psi'.(\Psi \otimes \Psi', P, Q) \in \mathcal{R}$ (extension of arbitrary assertion)*
3. *If $\Psi \rhd P \xrightarrow{\alpha} P'$ and $\mathrm{bn}(\alpha) \# \Psi, Q$, then there exists Q' such that $\Psi \rhd Q \xrightarrow{\alpha} Q'$ and $(\Psi, P', Q') \in \mathcal{R}$ (simulation)*

We let bisimilarity $\dot{\sim}$ *be the largest bisimulation. We write $P \dot{\sim}_\Psi Q$ to mean $(\Psi, P, Q) \in \dot{\sim}$, and $P \dot{\sim} Q$ for $P \dot{\sim}_{\mathbf{1}} Q$*

Clause 3 is the same as for pi-calculus bisimulation. Clause 1 requires that two bisimilar processes expose statically equivalent assertion environments. Clause 2 states that if two processes are bisimilar in an environment, they must be bisimilar in every extension of that environment. Without this clause, bisimulation is not preserved by parallel composition.

This definition might raise some red flags for the experienced concurrency theorist. We allow the matching transition from Q to have any provenance, irrespectively of what P's provenance is. Hence the Com rule uses information that is ignored for the purposes of bisimulation, which in most cases would result in a bisimilarity that is not preserved by the parallel operator.

Before showing that bisimilarity is nonetheless compositional, we will argue that bisimilarity would be too strong if Clause 4 required transitions with matching provenances. Consider two distinct terms M, N that are connected to the same channels; that is, for all Ψ, K we have $\Psi \vdash M \dot{\to} K$ iff $\Psi \vdash N \dot{\to} K$. We would expect $\overline{M}.0$ and $\overline{N}.0$ to be bisimilar because they offer the same interaction possibilities. With our definition, they are. But if bisimulation cared about provenance they would be distinguished, because transitions originating from $\overline{M}.0$ will have provenance M while those from $\overline{N}.0$ will have N.

The key intuition is that what matters is not which provenance a transition has, but which channels the provenance is connected to. The latter is preserved by Clause 3, as this key technical lemma—formally proven in Isabelle, by a routine induction—hints at:

Lemma 1. *(Find connected provenance)*

1. *If $\Psi \rhd P \xrightarrow[\pi]{\underline{M}\, N} P'$ and C is finitely supported, then there exists $\widetilde{b}_P, \Psi_P, \widetilde{x}, K$ such that $\mathcal{F}(P) = (\nu \widetilde{b}_P)\Psi_P$ and $\pi = (\nu \widetilde{b}_P; \widetilde{x})K$ and $\widetilde{b}_P \# \Psi, P, M, N, P', C, \widetilde{x}$ and $\widetilde{x} \# \Psi, P, N, P', C$ and $\Psi \otimes \Psi_P \vdash M \dot{\to} K$.*

2. A similar property for output transitions (elided).

In words, the provenance of a transition is always connected to its subject, and the frame binders can always be chosen sufficiently fresh for any context. This simplifies the proof that bisimilarity is preserved by parallel: in the original psi-calculi, one of the more challenging aspects of this proof is finding sufficiently fresh subjects to use in the COM-OLD rule, and then using associativity and symmetry to connect them (cf. [1, Lemma 5.11]). By Lemma 1 we already have a sufficiently fresh subject: our communication partner's provenance.

Theorem 1 (Congruence properties of strong bisimulation).

1. $P \mathrel{\dot{\sim}_\Psi} Q \quad \Rightarrow \quad P \mid R \mathrel{\dot{\sim}_\Psi} Q \mid R$
2. $P \mathrel{\dot{\sim}_\Psi} Q \quad \Rightarrow \quad (\nu x)P \mathrel{\dot{\sim}_\Psi} (\nu x)Q$ *if* $x\#\Psi$
3. $P_G \mathrel{\dot{\sim}_\Psi} Q_G \quad \Rightarrow \quad !P_G \mathrel{\dot{\sim}_\Psi} !Q_G$
4. $\forall i.P_i \mathrel{\dot{\sim}_\Psi} Q_i \quad \Rightarrow \quad \mathbf{case}\ \widetilde{\varphi} : \widetilde{P} \mathrel{\dot{\sim}_\Psi} \mathbf{case}\ \widetilde{\varphi} : \widetilde{Q}$ *if* $\widetilde{P}, \widetilde{Q}$ *are guarded*
5. $P \mathrel{\dot{\sim}_\Psi} Q \quad \Rightarrow \quad \overline{M}\,N.P \mathrel{\dot{\sim}_\Psi} \overline{M}\,N.Q$

Theorem 2 (Algebraic laws of strong bisimulation).

$$P \mathrel{\dot{\sim}_\Psi} P \mid \mathbf{0} \qquad P \mid (Q \mid R) \mathrel{\dot{\sim}_\Psi} (P \mid Q) \mid R \qquad P \mid Q \mathrel{\dot{\sim}_\Psi} Q \mid P \qquad (\nu a)\mathbf{0} \mathrel{\dot{\sim}_\Psi} \mathbf{0}$$

$$P \mid (\nu a)Q \mathrel{\dot{\sim}_\Psi} (\nu a)(P \mid Q) \text{ if } a\#P \qquad \overline{M}\,N.(\nu a)P \mathrel{\dot{\sim}_\Psi} (\nu a)\overline{M}\,N.P \text{ if } a\#M,N$$

$$\underline{M}(\lambda\widetilde{x})N.(\nu a)P \mathrel{\dot{\sim}_\Psi} (\nu a)\underline{M}(\lambda\widetilde{x})N.P \text{ if } a\#\widetilde{x},M,N \qquad !P \mathrel{\dot{\sim}_\Psi} P \mid !P$$

$$\mathbf{case}\ \widetilde{\varphi} : \widetilde{(\nu a)P} \mathrel{\dot{\sim}_\Psi} (\nu a)\mathbf{case}\ \widetilde{\varphi} : \widetilde{P} \text{ if } a\#\widetilde{\varphi} \qquad (\nu a)(\nu b)P \mathrel{\dot{\sim}_\Psi} (\nu b)(\nu a)P$$

The proofs of Theorems 1 and 2 have been mechanised in Nominal Isabelle. Note that bisimilarity is not preserved by input, for the same reasons as the pi-calculus. As in the pi-calculus, we can define *bisimulation congruence* as the substitution closure of bisimilarity, and thus obtain a true congruence which satisfies all the algebraic laws above. We have verified this in Nominal Isabelle, following [1].

The fact that bisimilarity is compositional yet ignores provenances suggests that the semantics could be reformulated without provenance annotations on labels. To achieve this, what is needed is a side-condition S for the COM rule which, given an input and an output with subjects M, K, determines if the input transition could have been derived from prefix K, and vice versa:

$$\frac{\Psi_Q \otimes \Psi \rhd P \xrightarrow{\overline{M}\,(\nu\widetilde{a})N} P' \qquad \Psi_P \otimes \Psi \rhd Q \xrightarrow{\underline{K}\,N} Q' \qquad S}{\Psi \rhd P \mid Q \xrightarrow{\tau} (\nu\widetilde{a})(P' \mid Q')}$$

But we already have such an S: the semantics *with* provenances! So we can let

$$S = \Psi_Q \otimes \Psi \rhd P \xrightarrow[(\nu\widetilde{b}_P;\widetilde{x})K]{\overline{M}(\nu\widetilde{a})N} P' \wedge \Psi_P \otimes \Psi \rhd Q \xrightarrow[(\nu\widetilde{b}_Q;\widetilde{y})M]{\underline{K}\,N} Q'$$

Of course, this definition is not satisfactory: the provenances are still there, just swept under the carpet. Worse, we significantly complicate the definitions by effectively introducing a stratified semantics. Thus the interesting question is not whether such an S exists (it does), but whether S can be formulated in a way that is significantly simpler than the semantics with provenances. The author believes the answer is negative: S is a property about the roots of the proof trees used to derive the transitions from P and Q. The provenance records just enough information about the proof trees to show that M and K are connected; with no provenances, it is not clear how this information could be obtained without essentially reconstructing the proof tree.

3.2 Validation

We have defined semantics and bisimulation, and showed that bisimilarity satisfies the expected laws. But how do we know that they are the right semantics, and the right bisimilarity? This section provides two answers to this question. First, we show that our developments constitute a conservative extension of the original psi-calculi. Second, we define a reduction semantics and barbed bisimulation that are in agreement with our (labelled) semantics and (labelled) bisimilarity.

Let $\longrightarrow_o$ and $\dot{\sim}_o$ denote semantics and bisimilarity as defined by Bengtson et al. [1], i.e., without provenances and with the Com-Old rule discussed in Section 2. The following result has been mechanised in Nominal Isabelle:

Theorem 3 (Conservativity). *When $\dot{\rightarrow}$ is symmetric and transitive we have $\dot{\sim}_o = \dot{\sim}$ and $\longrightarrow_o = \longrightarrow$.*

Our reduction semantics departs from standard designs [3,17] by relying on reduction contexts [7] instead of structural rules, for two reasons. First, standard formulations tend to include rules like these:

$$\frac{P \longrightarrow P'}{P \mid Q \longrightarrow P' \mid Q} \qquad \frac{}{\alpha.P + Q \mid \overline{\alpha}.R + S \longrightarrow P \mid R}$$

A parallel rule like the above would be unsound because Q might contain assertions that retract some conditions needed to derive P's reduction. The reduction axiom assumes prefix-guarded choice. We want our semantics to apply to the full calculus, without limiting the syntax to prefix-guarded **case** statements.

But first, a few auxiliary definitions. The *reduction contexts* are the contexts in which communicating processes may occur:

Definition 10 (Reduction contexts). *The* reduction contexts*, ranged over by C, are generated by the grammar*

$$\begin{array}{lll} C := P_G & \textit{(process)} & [\,] \qquad\qquad\qquad\qquad\qquad\qquad \textit{(hole)} \\ \quad C \mid C & \textit{(parallel)} & \textbf{case } \widetilde{\varphi} : \widetilde{P_G} \,[\!]\, \varphi' : C \,[\!]\, \widetilde{\varphi''} : \widetilde{Q_G} \quad \textit{(case)} \end{array}$$

$$\textsc{Struct}\ \frac{P \equiv Q \qquad Q \longrightarrow Q' \qquad Q' \equiv P'}{P \longrightarrow P'} \qquad \textsc{Scope}\ \frac{P \longrightarrow Q}{(\nu a)P \longrightarrow (\nu a)Q}$$

$$\textsc{Ctxt}\ \frac{\widetilde{\Psi} \vdash M \doteq N \qquad K = L[\widetilde{x} := \widetilde{T}] \qquad \forall \varphi \in \mathrm{conds}(C).\ \widetilde{\Psi} \vdash \varphi}{\widetilde{(\!|\Psi|\!)} \mid C[\overline{M}\, K.P,\ \underline{N}(\lambda \widetilde{x})L.Q] \longrightarrow \widetilde{(\!|\Psi|\!)} \mid P \mid Q[\widetilde{x} := \widetilde{T}] \mid \mathrm{ppr}(C)}$$

Table 2. Reduction semantics. Here $\widetilde{\Psi}$ abbreviates the composition $\Psi_1 \otimes \Psi_2 \otimes \ldots$, and $\widetilde{(\!|\Psi|\!)}$ abbreviates the parallel composition $(\!|\Psi_1|\!) \mid (\!|\Psi_2|\!) \mid \ldots$—for empty sequences they are taken to be $\mathbf{1}$ and $\mathbf{0}$ respectively.

Let $H(C)$ denote the number of holes in C. $C[\widetilde{P_G}]$ denotes the process that results from filling each hole of C with the corresponding element of $\widetilde{P_G}$, where holes are numbered from left to right; if $H(C) \neq |\widetilde{P_G}|$, $C[\widetilde{P_G}]$ is undefined.

We let *structural congruence* $\equiv$ be the smallest equivalence relation on processes derivable using Theorems 1 and 2. The *conditions* $\mathrm{conds}(C)$ and *parallel processes* $\mathrm{ppr}(C)$ of a context C are, respectively, the conditions in C that guard the holes, and the processes of C that are parallel to the holes:

$$\mathrm{ppr}(P_G) = P_G \qquad \mathrm{ppr}([\,]) = \mathbf{0} \qquad \mathrm{ppr}(C_1 \mid C_2) = \mathrm{ppr}(C_1) \mid \mathrm{ppr}(C_2)$$

$$\mathrm{ppr}(\mathbf{case}\ \widetilde{\varphi} : \widetilde{P_G} \,[]\, \varphi' : C \,[]\, \widetilde{\varphi''} : \widetilde{Q_G}) = \mathrm{ppr}(C) \qquad \mathrm{conds}(P_G) = \emptyset$$

$$\mathrm{conds}([\,]) = \emptyset \qquad \mathrm{conds}(C_1 \mid C_2) = \mathrm{conds}(C_1) \cup \mathrm{conds}(C_2)$$

$$\mathrm{conds}(\mathbf{case}\ \widetilde{\varphi} : \widetilde{P_G} \,[]\, \varphi' : C \,[]\, \widetilde{\varphi''} : \widetilde{Q_G}) = \{\varphi'\} \cup \mathrm{conds}(C)$$

Definition 11 (Reduction semantics). *The reduction relation $\longrightarrow \subseteq \mathbf{P} \times \mathbf{P}$ is defined inductively by the rules of Table 2.*

In words, Ctxt states that if an input and output prefix occur in a reduction context, we may derive a reduction if the following holds: the prefixes are connected in the current assertion environment, the message matches the input pattern, and all conditions guarding the prefixes are entailed by the environment. The $\mathrm{ppr}(C)$ in the reduct makes sure any processes in parallel to the holes are preserved.

Theorem 4. *$P \longrightarrow P'$ iff there is P'' such that $\mathbf{1} \rhd P \xrightarrow{\tau} P''$ and $P'' \equiv P'$*

For barbed bisimulation, we need to define what the observables are, and what contexts an observer may use. We follow previous work by Johansson et al. [15] on weak barbed bisimilarity for the original psi-calculi on both counts. First, we take the barbs to be the output labels a process can exhibit: we define

$P \downarrow_{\overline{M}(\nu\widetilde{a})N}$ (P exposes $\overline{M}(\nu\widetilde{a})N$) to mean $\exists P'.\ 1 \rhd P \xrightarrow{\overline{M}(\nu\widetilde{a})N} P'$. We write $P \downarrow_{\overline{M}}$ for $\exists \widetilde{a}, N.P \downarrow_{\overline{M}(\nu\widetilde{A})N}$, and $P \Downarrow_\alpha$ for $P \xrightarrow{\tau}{}^{\star} \downarrow_\alpha$. Second, we let observers use *static* contexts, i.e. ones built from parallel and restriction.

Definition 12 (Barbed bisimilarity). Barbed bisimilarity*, written* $\dot{\sim}_{\text{barb}}$*, is the largest equivalence on processes such that* $P \dot{\sim}_{\text{barb}} Q$ *implies*

1. *If* $P \downarrow_{\overline{M}(\nu\widetilde{a})N}$ *and* $\widetilde{a}\#Q$ *then* $Q \downarrow_{\overline{M}(\nu\widetilde{a})N}$ *(barb similarity)*
2. *If* $P \longrightarrow P'$ *then there exists* Q' *such that* $Q \longrightarrow Q'$ *and* $P' \dot{\sim}_{\text{barb}} Q'$ *(reduction simulation)*
3. $(\nu\widetilde{a})(P \mid R) \dot{\sim}_{\text{barb}} (\nu\widetilde{a})(Q \mid R)$ *(closure under static contexts)*

Our proof that barbed and labelled bisimilarity coincides only considers psi-calculi with a certain minimum of sanity and expressiveness. This rules out some degenerate cases: psi-calculi where there are messages that can be sent but not received, and psi-calculi where no transitions whatsoever are possible.

Definition 13. *A psi-calculus is* observational *if:*

1. *For all* P *there are* M_P, K_P *such that* $\mathcal{F}(P) \vdash M_P \dot{\to} K_P$ *and not* $P \Downarrow_{\overline{K_p}}$.
2. *If* $N = (\widetilde{x}\ \widetilde{y}) \cdot M$ *and* $\widetilde{y}\#M$ *and* $\widetilde{x}, \widetilde{y}$ *are distinct then* $M[\widetilde{x} := \widetilde{y}] = N$.

The first clause means that no process can exhaust the set of barbs. Hence observing contexts can signal success or failure without interference from the process under observation. For example, in the pi-calculus M_P, K_P can be any name x such that $x\#P$. The second clause states that for swapping of distinct names, substitution and permutation have the same behaviour. Any standard definition of simultaneous substitution should satisfy this requirement. These assumptions are present, explicitly or implicitly, in the work of Johansson et al. [15]. Ours are given a slightly weaker formulation.

We can now state the main result of this section:

Theorem 5. *In all observational psi-calculi,* $P \dot{\sim}_{\text{barb}} Q$ *iff* $P \dot{\sim}_1 Q$.

4 Expressiveness

In this section, we study two examples of the expressiveness gained by dropping symmetry and transitivity.

4.1 Pi-calculus with preorders

Recall that pi-F [25] extends the pi-calculus with name equalities ($x = y$) as first-class processes. Communication in pi-F gives rise to equalities rather than substitutions, so e.g. $xy.P \mid \overline{x}z.Q$ reduces to $y = z \mid P \mid Q$: the input and output

objects are fused. Hirschkoff et al. [11] observed that fusion and subtyping are fundamentally incompatible, and propose a generalisation of pi-F called the *pi-calculus with preorders* or πP to resolve the issue.

We are interested in πP because its channel connectivity is not transitive. The equalities of pi-F are replaced with *arcs* a/b ("a is above b") which act as one-way fusions: anything that can be done with b can be done with a, but not the other way around. The effect of a communication is to create an arc with the output subject above the input subject, so $x(y).P \mid \overline{x}(z).Q$ reduces to $(\nu xy)(z/y \mid P \mid Q)$. We write $\prec$ for the reflexive and transitive closure of the "is above" relation. Two names x, y are considered *joinable* for the purposes of synchronisation if some name z is above both of them: formally, we write $x \curlyvee y$ for $\exists z.x \prec z \wedge y \prec z$.

Hirschkoff et al. conclude by saying that "[it] could also be interesting to study the representation of πP into Psi-calculi. This may not be immediate because the latter make use of on an equivalence relation on channels, while the former uses a preorder" [11, p. 387]. Having lifted the constraint that channels form an equivalence relation, we happily accept the challenge. We write ΨP for the psi-calculus we use to embed πP. We follow the presentation of πP from [12,13], where the behavioural theory is most developed.

Definition 14. *The psi-calculus ΨP is defined with the following parameters:*

$$\mathbf{T} \triangleq \mathcal{N} \qquad \mathbf{C} \triangleq \{x \prec y : x, y \in \mathcal{N}\} \cup \{x \curlyvee y : x, y \in \mathcal{N}\}$$

$$\mathbf{A} \triangleq \mathcal{P}_{fin}(\{x \prec y : x, y \in \mathcal{N}\}) \qquad \mathbf{1} \triangleq \{\} \qquad \otimes \triangleq \cup$$

$$\dot{\rightarrow} \triangleq \curlyvee \qquad \vdash \triangleq \textit{the relation denoted} \vdash \textit{in [13].}$$

The prefix operators of πP are different from those of psi-calculi: objects are always bound, communication gives rise to an arc rather than a substitution, and a conditional silent prefix $[\varphi]\tau.P$ is included.[2] These are encodable as follows:

Definition 15 (Encoding of prefixes). *The encoding $[\![\_]\!]$ from πP to ΨP is homomorphic on all operators except prefixes and arcs, where it is defined by*

$$[\![a/b]\!] = (\!|b \prec a|\!) \qquad [\![\overline{a}(y).P]\!] = (\nu xy)(\overline{a}x.((\!|x \prec y|\!) \mid [\![P]\!])) \textit{ where } x\#y, P$$

$$[\![a(y).P]\!] = (\nu y)(\underline{a}(\lambda x)x.((\!|y \prec x|\!) \mid [\![P]\!])) \textit{ where } x\#y, P$$

$$[\![[\varphi]\tau.P]\!] = \mathbf{case}\ \varphi : (\nu x)(\underline{x}(\lambda x)x.0 \mid \overline{x}x.[\![P]\!]) \textit{ where } x\#P$$

This embedding of πP in psi-calculi comes with a notion of bisimilarity per Definition 9. We show that it coincides with the labelled bisimilarity for πP (written $\sim$) introduced in [12,13].

Theorem 6. *$P \sim Q$ iff $[\![P]\!] \dot{\sim} [\![Q]\!]$*

[2] We ignore protected prefixes because they are redundant, cf. Remark 1 of [12].

Thus our encoding validates the behavioural theory of πP by connecting it to our fully mechanised proofs, while also showing that a substantially different design of the LTS yields the same bisimilarity. We will briefly compare these designs. While we do rewriting of subjects in the prefix rules, Hirschkoff et al. instead use relabelling rules like this one (mildly edited to match our notation):

$$\frac{P \xrightarrow{a(x)} P' \qquad \mathcal{F}(P) \vdash a \prec b}{P \xrightarrow{b(x)} P'}$$

An advantage of this rule is that it allows input and output labels to be as simple as pi-calculus labels. A comparative disadvantage is that it is not syntax-directed, and that the LTS has more rules in total. Note that this rule would not be a viable alternative to provenances in psi-calculi: since it can be applied more than once in a derivation, its inclusion assumes that the channels form a preorder wrt. connectivity.

πP also has labels $[\varphi]\tau$, meaning that a silent transition is allowed in environments where φ is true. A rule for rewriting φ to a weaker condition, similar to the above rule for subject rewriting, is included. Psi-calculi does not need this because the PAR rules take the assertion environment into account. πP transitions of kind $P \xrightarrow{[\varphi]\tau} P'$ correspond to ΨP transitions of kind $\{\varphi\} \rhd P \xrightarrow{\tau} P'$.

Interestingly, the analogous full abstraction result fails to hold for the embedding of pi-F in psi-calculi by Bengtson et al. [1], because outputs that emit distinct but fused names are distinguished by psi-calculus bisimilarity. This issue does not arise here because πP objects are always bound; however, we believe the encoding of Bengtson et al. can be made fully abstract by encoding free output with bound output, exploiting the pi-F law $a\,y.Q \sim a(x)(Q \mid x = y)$.

4.2 Mixed choice

This section will argue that because we allow non-transitive channel connectivity, the **case** operator of psi-calculi becomes superfluous. The formal results here will focus on encoding the special case of mixed choice. We will then briefly discuss how to generalise these results to the full **case** operator.

Choice, written $P + Q$, is a process that behaves as either P or Q. In psi-calculi we consider $P + Q$ to abbreviate $\mathbf{case}\ \top : P \,[]\, \top : Q$ for some condition $\top$ that is always entailed. *Mixed choice* means that in $P + Q$, P and Q must be prefix-guarded; that is, the outermost operators of P, Q must be input or output prefixes. In particular, mixed choice allows choice between an input and an output. There is a straightforward generalisation to n-ary sums that, in order to simplify the presentation, we will not consider here.

Fix a psi-calculus $\mathcal{P} = (\mathbf{T}, \mathbf{A}, \mathbf{C}, \vdash, \otimes, \mathbf{1}, \dot{\to})$ with mixed choice; this will be our source language. We will construct a target psi-calculus and an encoding such that the target terms make no use of the **case** operator. The target language $\mathcal{E}(\mathcal{P})$ adds to $\mathbf{T}$ the ability to tag a term M with a name x; we write M_x for the tagged term. We write α_x for tagging the subject of the prefix α with x. Tags are

used to uniquely identify which choice statement a prefix is a summand of. As the assertions of $\mathcal{E}(\mathcal{P})$ we use $\mathbf{A} \times \mathcal{P}_{\text{fin}}(\mathcal{N})$, where $\mathcal{P}_{\text{fin}}(\mathcal{N})$ are the *disabled tags*.

The encoding $[\![\_]\!]$ from $\mathcal{P}$ to $\mathcal{E}(\mathcal{P})$ is homomorphic on all operators except assertion and choice, where it is defined as follows:

$$[\![(\!|\Psi|\!)]\!] = (\!|(\Psi, \emptyset)|\!) \qquad [\![\alpha.P + \beta.Q]\!] = (\nu x)(\alpha_x.([\![P]\!] \mid (\!|(\mathbf{1}, \{x\})|\!) \mid \beta_x.([\![Q]\!] \mid (\!|(\mathbf{1}, \{x\})|\!)$$

where $x \# \alpha, \beta, P, Q$. If we disregard the tag x, we see that the encoding simply offers up both summands in parallel. This clearly allows all behaviours of $\alpha.P + \beta.Q$, but there are two additional behaviours we must prevent: (1) communication between the summands, and (2) lingering summands firing after the other branch has already been taken. The tagging mechanism prevents both, as a consequence of how we define channel equivalence on tagged terms in $\mathcal{E}(\mathcal{P})$:

$$(\Psi, \mathbf{N}) \vdash M_x \dot{\leftrightarrow} N_y \quad \text{if } \Psi \vdash M \leftrightarrow N \text{ and } x \neq y \text{ and } x, y \notin \mathbf{N}$$

That is, tagged channels are connected if the underlying channel is connected. To prevent (1) we require the tags to be different, and to prevent (2) we require that the tags are not disabled. Note that this channel connectivity is not transitive, not reflexive, and not monotonic wrt. assertion composition—not even if the source language connectivity is.

Theorem 7 (Correctness of choice encoding).

1. *If* $\Psi \rhd P \xrightarrow{\alpha} P'$ *then there is* P'' *such that* $(\Psi, \emptyset) \rhd [\![P]\!] \xrightarrow{\alpha} P''$ *and* $P'' \dot{\sim}_{(\Psi,\emptyset)} [\![P']\!]$.
2. *If* $(\Psi, \emptyset) \rhd [\![P]\!] \xrightarrow{\alpha} P'$ *then there is* P'' *such that* $\Psi \rhd P \xrightarrow{\alpha_\perp} P''$ *and* $P' \dot{\sim}_{(\Psi,\emptyset)} [\![P'']\!]$.
3. $P \dot{\sim}_{\mathbf{1}} Q$ *iff* $[\![P]\!] \dot{\sim}_{(\mathbf{1},\emptyset)} [\![Q]\!]$.

Here $\alpha_\perp$ denote the label α with all tags removed. It is immediate from Theorem 7 and the definition of $[\![\_]\!]$ that our encoding also satisfies the other standard quality criteria [10]: it is compositional, it is name invariant, and it preserves and reflects barbs and divergence.

In the original psi-calculi, our target language is invalid because of non-transitive connectivity. If we remove the requirement that tags are distinct, and only allow *separate* choice (where either both summands are inputs or both summands are outputs), the encoding is correct for the original psi-calculi.

These results generalise in a straightforward way to mixed Case statements **case** $\varphi_1 : \alpha.P \,[]\, \varphi_2 : \beta.Q$ by additionally tagging terms with a condition, i.e. M_{x,φ_1}, that must be entailed in order to derive connectivity judgements involving the term. The generalisation to free choice, i.e. $P + Q$ where P, Q can be anything, is more involved and sacrifices some compositionality. The idea is to use sequences of tags, representing which branches of which (possibly nested) case statements a prefix can be found in, and disallowing communication between prefixes in distinct branches of the same Case operator.

5 Conclusion and related work

We have seen how psi-calculi can be conservatively extended to allow asymmetric and non-transitive communication topologies, sacrificing none of the bisimulation meta-theory. This confers enough expressiveness to capture a pi-calculus with preorders, and makes mixed choice a derived operator.

The work of Hirschkoff et al. [11] is closely related in that it uses non-transitive connectivity; see Section 4.1 for an extensive discussion.

Broadcast psi-calculi [5] extend psi-calculi with broadcast communication in addition to point-to-point communication. There, point-to-point channels must still be symmetric and transitive, but for broadcast channels this condition is lifted, at the cost of introducing other side-conditions on how names are used: broadcast prefixes must be connected via intermediate *broadcast channels* which have no greater support than either of the prefixes it connects, precluding language features such as name fusion. We believe provenances could be used to define a version of broadcast psi-calculi that does not need this side-condition.

Kouzapas et al. [16] define a similar reduction context semantics for (broadcast) psi-calculi. Their reduction contexts requires three kinds of numbered holes with complicated side-conditions on how the holes may be filled; we have attempted to simplify the presentation by having only one kind of hole. While (weak) barbed congruence for psi-calculi has been studied before [15] (see Section 3.2), barbed congruence was defined in terms of the labelled semantics rather than a reduction semantics, thus weakening its claim to independent confirmation slightly.

There is a rich literature on choice encodings for the pi-calculus [10,20,21,22,23], with many separation and encodability results under different quality criteria for different flavours of choice. Encodings typically require complicated protocols and tradeoffs between quality criteria. Thanks to the greater expressive power of psi-calculi, our encoding is simpler and satisfies stronger quality criteria than any choice encoding for the pi-calculus. Closest to ours is the choice encoding of CCS into the DiX calculus by Busi and Gorrieri [6]. DiX introduces a primitive for annotating processes with *conflict sets*, that are intended as a generalisation of choice. Processes with overlapping conflict sets cannot interact, and when a process acts, every process with an overlapping conflict set is killed. These conflict sets perform the same role in the encoding as our tags do. We believe the tagging scheme used in our choice encoding also captures DiX-style conflict sets.

Acknowledgements

These ideas have benefited from discussions with many people at Uppsala University, ITU Copenhagen, the University of Oslo and Data61/CSIRO, including Jesper Bengtson, Christian Johansen, Magnus Johansson and Joachim Parrow. I would also like to thank Jean-Marie Madiot and the anonymous reviewers for valuable comments on earlier versions of the paper.

References

1. Bengtson, J., Johansson, M., Parrow, J., Victor, B.: Psi-calculi: A framework for mobile processes with nominal data and logic. Logical Methods in Computer Science **7**(1) (2011). https://doi.org/10.2168/LMCS-7(1:11)2011
2. Bengtson, J., Parrow, J., Weber, T.: Psi-calculi in Isabelle. J. Autom. Reasoning **56**(1), 1–47 (2016). https://doi.org/10.1007/s10817-015-9336-2
3. Berry, G., Boudol, G.: The chemical abstract machine. In: Proceedings of the 17th ACM SIGPLAN-SIGACT symposium on Principles of programming languages. pp. 81–94. POPL '90, ACM, New York, NY, USA (1990). https://doi.org/10.1145/96709.96717
4. Borgström, J., Gutkovas, R., Parrow, J., Victor, B., Pohjola, J.Å.: A sorted semantic framework for applied process calculi (extended abstract). In: TGC. pp. 103–118 (2013). https://doi.org/10.1007/978-3-319-05119-2_7
5. Borgström, J., Huang, S., Johansson, M., Raabjerg, P., Victor, B., Åman Pohjola, J., Parrow, J.: Broadcast psi-calculi with an application to wireless protocols. Software and System Modeling **14**(1), 201–216 (2015). https://doi.org/10.1007/s10270-013-0375-z
6. Busi, N., Gorrieri, R.: Distributed conflicts in communicating systems. In: Ciancarini, P., Nierstrasz, O., Yonezawa, A. (eds.) Object-Based Models and Languages for Concurrent Systems, ECOOP'94 Workshop on Models and Languages for Coordination of Parallelism and Distribution, Bologna, Italy, July 5, 1994, Selected Papers. Lecture Notes in Computer Science, vol. 924, pp. 49–65. Springer (1994). https://doi.org/10.1007/3-540-59450-7_4
7. Felleisen, M., Hieb, R.: The revised report on the syntactic theories of sequential control and state. Theor. Comput. Sci. **103**(2), 235–271 (1992). https://doi.org/10.1016/0304-3975(92)90014-7
8. Gabbay, M.J., Pitts, A.M.: A new approach to abstract syntax with variable binding. Formal Aspects of Computing **13**, 341–363 (2002). https://doi.org/10.1007/s001650200016
9. Gardner, P., Wischik, L.: Explicit fusions. In: Nielsen, M., Rovan, B. (eds.) Proceedings of MFCS 2000. vol. 1893, pp. 373–382 (2000). https://doi.org/10.1007/3-540-44612-5_33
10. Gorla, D.: Towards a unified approach to encodability and separation results for process calculi. Inf. Comput. **208**(9), 1031–1053 (2010). https://doi.org/10.1016/j.ic.2010.05.002
11. Hirschkoff, D., Madiot, J., Sangiorgi, D.: Name-passing calculi: From fusions to preorders and types. In: 28th Annual ACM/IEEE Symposium on Logic in Computer Science, LICS 2013, New Orleans, LA, USA, June 25-28, 2013. pp. 378–387. IEEE Computer Society (2013). https://doi.org/10.1109/LICS.2013.44
12. Hirschkoff, D., Madiot, J., Xu, X.: A behavioural theory for a π-calculus with preorders. J. Log. Algebr. Meth. Program. **84**(6), 806–825 (2015). https://doi.org/10.1016/j.jlamp.2015.07.001
13. Hirschkoff, D., Madiot, J., Xu, X.: A behavioural theory for a π-calculus with preorders. In: Dastani, M., Sirjani, M. (eds.) Fundamentals of Software Engineering - 6th International Conference, FSEN 2015 Tehran, Iran, April 22-24, 2015, Revised Selected Papers. Lecture Notes in Computer Science, vol. 9392, pp. 143–158. Springer (2015). https://doi.org/10.1007/978-3-319-24644-4_10
14. Johansson, M.: Psi-calculi: a framework for mobile process calculi: Cook your own correct process calculus - just add data and logic. Ph.D. thesis, Uppsala University, Division of Computer Systems (2010)

15. Johansson, M., Bengtson, J., Parrow, J., Victor, B.: Weak equivalences in psi-calculi. In: LICS. pp. 322–331. IEEE Computer Society (2010). https://doi.org/10.1109/LICS.2010.30
16. Kouzapas, D., Gutkovas, R., Gay, S.J.: Session types for broadcasting. In: Donaldson, A.F., Vasconcelos, V.T. (eds.) Proceedings 7th Workshop on Programming Language Approaches to Concurrency and Communication-cEntric Software, PLACES 2014, Grenoble, France, 12 April 2014. EPTCS, vol. 155, pp. 25–31 (2014). https://doi.org/10.4204/EPTCS.155.4
17. Milner, R.: Functions as processes. In: Proceedings of the seventeenth international colloquium on Automata, languages and programming. pp. 167–180. Springer-Verlag New York, Inc., New York, NY, USA (1990). https://doi.org/10.1007/BFb0032030
18. Milner, R., Parrow, J., Walker, D.: A calculus of mobile processes, part I/II. Inf. Comput. **100**(1), 1–77 (1992). https://doi.org/10.1016/0890-5401(92)90008-4
19. Nanz, S., Hankin, C.: A framework for security analysis of mobile wireless networks. Theor. Comput. Sci. **367**(1-2), 203–227 (2006)
20. Nestmann, U., Pierce, B.C.: Decoding choice encodings. Inf. Comput. **163**(1), 1–59 (2000). https://doi.org/10.1006/inco.2000.2868
21. Palamidessi, C.: Comparing the expressive power of the synchronous and the asynchronous pi-calculus. In: Lee, P., Henglein, F., Jones, N.D. (eds.) Conference Record of POPL'97: The 24th ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages, Papers Presented at the Symposium, Paris, France, 15-17 January 1997. pp. 256–265. ACM Press (1997). https://doi.org/10.1145/263699.263731
22. Peters, K., Nestmann, U.: Is it a "good" encoding of mixed choice? In: Birkedal, L. (ed.) Foundations of Software Science and Computational Structures - 15th International Conference, FOSSACS 2012, Held as Part of the European Joint Conferences on Theory and Practice of Software, ETAPS 2012, Tallinn, Estonia, March 24 - April 1, 2012. Proceedings. Lecture Notes in Computer Science, vol. 7213, pp. 210–224. Springer (2012). https://doi.org/10.1007/978-3-642-28729-9_14
23. Peters, K., Nestmann, U., Goltz, U.: On distributability in process calculi. In: Felleisen, M., Gardner, P. (eds.) Programming Languages and Systems - 22nd European Symposium on Programming, ESOP 2013, Held as Part of the European Joint Conferences on Theory and Practice of Software, ETAPS 2013, Rome, Italy, March 16-24, 2013. Proceedings. Lecture Notes in Computer Science, vol. 7792, pp. 310–329. Springer (2013). https://doi.org/10.1007/978-3-642-37036-6_18
24. Urban, C.: Nominal techniques in Isabelle/HOL. Journal of Automated Reasoning **40**(4), 327–356 (May 2008). https://doi.org/10.1007/s10817-008-9097-2
25. Wischik, L., Gardner, P.: Explicit fusions. Theoretical Computer Science **304**(3), 606–630 (2005). https://doi.org/10.1016/j.tcs.2005.03.017

Appendix

This appendix contains proofs of the formal claim made in the paper, as well as some comments on the accompanying Isabelle formalisation.

Notes on the formalisation

Regrettably, a discussion of the Isabelle formalisation did not fit within the page budget.

The Isabelle formalisation can be downloaded from `https://github.com/IlmariReissumies/newpsi`. The current version runs on Isabelle2018 using the HOL-Nominal logic. To start Isabelle with the HOL-Nominal logic loaded the command-line invocation is

```
isabelle jedit -l HOL-Nominal
```

The semantics and induction principles are defined in the theory file `Semantics`. Note that since frames in the formalisation take a type parameter — that is, `'a frame` extends the nominal datatype `'a` with a binding sequence — provenances are modelled by the type `'term frame frame option`. This corresponds to an arbitrary nominal datatype (the locale parameter representing terms) extended with an inner and outer binding sequence, and the additional value `None` which encodes $\bot$.

The requisites of psi-calculi are the various locale assumptions scattered throughout the theory files: see in particular `Subst_Term` for substitution laws, and `Frame` for monoid laws of $\otimes$. Confusingly, even though there are no assumptions on channel connectivity other than equivariance, for historical reasons it is still written with an Isabelle notation similar to $\leftrightarrow$, which suggest bidirectionality even though it is not in general symmetric.

Lemma 1 is the conjunction of `input_provenance` and `output_provenance` from `Semantics`.

The clauses of Theorem 1 are proven in the theory file `Bisim_Pres`, except for Theorem 1.3 which is is `Bisim_Struct_Cong.bisim_bang_pres`. The clauses of Theorem 2 can be found in the theory file `Bisim_Struct_Cong`.

Theorem 3 is the conjunction of the theorems `old_semantics_sound` and `old_semantics_complete` from `Old_Semantics`; and `old_bisim_eq_bisim` from `Old_Bisimulation`. Note that the precondition that connectivity is transitive and symmetric is present as locale assumptions in the locale `Old_Semantics.old_psi`. This locale contains a copy of Bengtson's original definition of the psi-calculi semantics, renamed so as to avoid naming conflicts with our new one.

Finally, the formalisation of bisimulation congruence alluded to in the paper resides in `Bisim_Subst`.

The formalisation uses Bengtson's formalisation of the original psi-calculi as a starting point. Almost all proofs are different in some way; in some cases for purely syntactic reasons (provenances must now be considered), but often

for more interesting reasons related to the loss of symmetry and transitivity, particularly in the congruence proofs for bisimilarity.

One of the most difficult parts of the old proofs was the case when a communication is to be derived between P with subject M and Q with subject K and $\Psi \otimes \Psi_P \otimes \Psi_Q \vdash M' \leftrightarrows K$, and Q is swapped for a bisimilar process R; In particular, this requires finding a subject M' such that $\Psi \otimes \Psi_P \otimes \Psi_R \vdash M' \leftrightarrows K$ and M' is fresh for $\mathcal{F}(P)$. This is proved to be possible by a laborious process in the *switching lemma* [1, Lemma 5.11], where the main thrust of the proof idea is that we can always choose as M' the prefix that R used to derive its transition. With our semantics, this exercise is unnecessary: by construction, P's subject is already Q's prefix, and R's prefix is already present in the provenance.

There is one spot where, interestingly, the introduction of provenances complicates proofs that were previously fairly routine: the proof that replication preserves bisimulation. Suppose $P \mathrel{\dot{\sim}} Q$ and $!P$ derives a τ transition from communication between two unfolded copies of P, with input subject M and output subject K. We need to mimic the same communication between two copies of Q, but after using $P \mathrel{\dot{\sim}} Q$ to match the input transition, the subject M is not useful to derive a communication since it is P's provenance, not Q's. In order to obtain eligible subjects M', K' we need to repeatedly apply Lemma 1, and relabel the transitions using `Semantics.comm1_aux` and `Semantics.comm2_aux`.

Pi-calculus with preorders

Readers of this section would do well to acquaint themselves with the work of Hirschkoff et al. [12,13], from which we will freely use definitions and theorems without restating them here. In particular, we will use α, β to range over protected names, and sometimes use the extension of $\vdash$ to protected names, even though the psi-calculus instance does not use them.

Definition 16. *The psi-calculus ΨP is defined with the following parameters:*

$$\mathbf{T} \triangleq \mathcal{N} \qquad \mathbf{C} \triangleq \{x \prec y : x, y \in \mathcal{N}\} \cup \{x \curlyvee y : x, y \in \mathcal{N}\} \cup \top$$

$$\mathbf{A} \triangleq \mathcal{P}_{fin}(\{x \prec y : x, y \in \mathcal{N}\}) \qquad \mathbf{1} \triangleq \{\} \qquad \otimes \triangleq \cup$$

$$\dot{\rightarrow} \triangleq \curlyvee \qquad \vdash \triangleq \textit{the relation denoted} \vdash \textit{in [13].}$$

$\top$ is outside the scope of the relation $\vdash$ in [13]; here we let $\Psi \vdash T$ hold for all Ψ.

Note that $\top$ is used to encode the choice of πP, as par the section on mixed choice. Technically this is just a convenience since we may use any reflexive arc as condition guard in place of $\top$, at the cost of either name invariance or the homomorphic translation of choice.

All statements about psi-calculi made in this section shall be taken to refer to the psi-calculus ΨP. We will sometimes overload Ψ as shorthand for a πP parallel composition $\Pi_{\varphi \in \Psi} \varphi$, where the φs may occur in any order.

The main result, that psi-calculus bisimilarity on encoded terms coincides with πP bisimilarity on source terms, is an immediate corollary of Lemmas 16–17.

Lemma 2. *ΨP is a valid psi-calculus.*

Proof. Commutativity, associativity and identity are immediate from the corresponding laws about $\cup$.

For compositionality, we know that $\Psi \simeq \Psi'$, and need to show that $\Psi \cup \Psi'' \vdash \varphi$ implies $\Psi \cup \Psi'' \vdash \varphi$ (the proof for the $\Leftarrow$ direction is the same). We proceed by induction on the derivation of the judgement $\Psi \cup \Psi' \vdash \varphi$. We have formalised this proof in Isabelle. All cases are trivial; in the base case of $\vdash_{\text{-IN}}$ where $\varphi \in \Psi \cup \Psi''$, we use static equivalence in the subcase where $\varphi \in \Psi$.

Lemma 3. *If $\forall \varphi \in \Psi'.\Psi \vdash \varphi$ and $\Psi' \vdash \varphi'$ then $\Psi \vdash \varphi'$.*

Proof. By induction on the derivation of $\Psi' \vdash \varphi'$. In the $\vdash_{\text{IN}}$ case, we have $\varphi' \in \Psi'$ and the desired result follows from the premise $\forall \varphi \in \Psi'.\Psi \vdash \varphi$.

Lemma 4 (Monotonitiy of $\vdash$). *If $\Psi \vdash \varphi$ then $\Psi \otimes \Psi' \vdash \varphi$.*

Proof. By induction on the derivation of $\Psi \vdash \varphi$.

Lemma 5. *If $\Psi \cup \Psi' \vdash \varphi$ and Ψ, Ψ' are finite then there exists finite Ψ_1, Ψ_1' such that Ψ_1, Ψ_1' and $\Psi \vdash \Psi_1$ and $\Psi' \vdash \Psi_1'$ and $\Psi_1 \cup \Psi_1' \vdash \varphi$ and $\mathsf{n}(\Psi_1) \cup \mathsf{n}(\Psi_1') \subseteq \mathsf{n}(\varphi) \cup (\mathsf{n}(\Psi) \cap \mathsf{n}(\Psi'))$.*

Proof. By induction on the derivation of the judgement $\Psi \cup \Psi' \vdash \varphi$. In the $\vdash_{\text{Trans}}$, $\vdash_{\text{Join}}$ and $\vdash_{\text{ExtJoin}}$ cases we use the fact that if $\Psi \vdash \varphi$ and Ψ is finite and φ is not reflexive then $\mathsf{n}(\varphi) \subseteq \mathsf{n}(\Psi)$.

Lemma 6. *$\mathcal{F}([\![P]\!]) \vdash \varphi$ iff $P \rhd \varphi$*

Proof. $\Leftarrow$ By induction on the derivation of $P \rhd \varphi$. In the $\rhd_{Par-L}$ and $\rhd_{Par-R}$ cases we use monotonicity of $\vdash$. In the $\rhd_{\text{Combine}}$ case, we have $P \rhd \Psi$ and $\Psi \vdash \varphi$; by the induction hypothesis we have $\forall \varphi' \in \Psi.\mathcal{F}([\![P]\!]) \vdash \varphi'$. Let $\mathcal{F}([\![P]\!]) = (\nu \widetilde{b}_{[\![P]\!]})\Psi_{[\![P]\!]}$ where $\widetilde{b}_{[\![P]\!]} \# \Psi, \varphi, P$. By definition of $\vdash$ on frames we have $\Psi_{[\![P]\!]} \vdash \varphi'$ for all $\varphi' \in \Psi'$. By Lemma 3 we get $\Psi_{[\![P]\!]} \vdash \varphi$; $\mathcal{F}([\![P]\!]) \vdash \varphi$ follows by definition of frame entailment.

$\Rightarrow$ By induction on P. The interesting case is parallel composition, where $P = Q \mid R$. Fix frames $\mathcal{F}(Q) = (\nu \widetilde{b}_Q)\Psi_Q$ and $\mathcal{F}(R) = (\nu \widetilde{b}_R)\Psi_R$ such that $\widetilde{b}_Q \# \Psi_R, \widetilde{b}_R, Q, R, \varphi$ and $\widetilde{b}_R \# \Psi_Q, \widetilde{b}_Q, Q, R, \varphi$. Since φ is sufficiently fresh we have $\Psi_Q \cup \Psi_R \vdash \varphi$. By Lemma 5 we obtain finite Ψ, Ψ' such that $\Psi \otimes \Psi' \vdash \varphi$ and $\Psi_Q \vdash \Psi'$ and $\Psi_R \vdash \Psi'$ and $\mathsf{n}(\Psi) \cup \mathsf{n}(\Psi') \subseteq \mathsf{n}(\varphi) \cup (\mathsf{n}(\Psi_Q) \cap \mathsf{n}(\Psi_R))$. By definition of freshness and the freshness assumptions of the frame binders $\widetilde{b}_Q, \widetilde{b}_R$ we have $\widetilde{b}_Q, \widetilde{b}_R \# \Psi, \Psi'$, which means these facts can be observed outside the frame binders: $\mathcal{F}(Q) \vdash \Psi$ and $\mathcal{F}(R) \vdash \Psi'$. By the induction hypothesis we obtain $Q \rhd \Psi$ and $R \rhd \Psi'$. By repeatedly applying $\rhd_{Par-L}$ and $\rhd_{Par-R}$ we get $Q \mid R \rhd \Psi \cup \Psi'$; by rule $\rhd_{Combine}$ it follows that $P = Q \mid R \rhd \varphi$.

Lemma 7. $(\!|y \prec y|\!) \mathrel{\dot\sim_\Psi} 0$

Proof. Easy since $\mathcal{F}(0) \vdash y \prec y$.

Lemma 8. *If $\Psi \cup a \prec b \vdash d \curlyvee b$ and $b \# \Psi$ and $d \neq b$ and Ψ is a finite set of $\prec$ axioms then $\Psi \vdash d \prec a$.*

Proof. $\Psi \cup b/a \vdash d \curlyvee b$ must have been inferred by a chain of non-reflexive $\prec$ judgements $a_0 \prec a_1 \prec \cdots \prec a_n$ and $b_0 \prec b_1 \prec \cdots \prec b_m$ such that $a_0 = d$, $b_0 = b$ and $a_n = b_m$ and each $a_i \prec a_{i+1}$ or $b_i \prec b_{i+1}$ is an element of $\Psi \cup a \prec b$. Since nothing is above b, $m = 0$ and $a_n = b$. Thus the arc from $a_{n-1} \prec a_n$ can only be $a \prec b$, which suffices.

Lemma 9. *If $\Psi \rhd P \xrightarrow[(\nu \widetilde{b}_P;\widetilde{z})b]{\underline{a}\, x} P'$ and $P \mathrel{\dot\sim_\Psi} Q$ and $\mathcal{F}(P) = (\nu \widetilde{b}_P)\Psi_P$ and $\widetilde{b}_P, \widetilde{z} \# \Psi, P, a, x, Q$ and $\widetilde{b}_P \# \widetilde{z}$ and $\Psi \cup \Psi_P \vdash b \prec a$ and*
$\mathcal{F}(Q) = (\nu \widetilde{b}_Q)\Psi_Q$ *and* $\widetilde{b}_Q \# \Psi, P, a, x, Q, \widetilde{b}_P, \widetilde{z}$
then there is $\widetilde{y}, c, Q'$ such that $\widetilde{y} \# \Psi, P, a, x, Q, \widetilde{b}_P, \widetilde{b}_Q$ and $\Psi \rhd Q \xrightarrow[(\nu \widetilde{b}_Q;\widetilde{y})c]{\underline{a}\, x} Q'$ and $P' \mathrel{\dot\sim_\Psi} Q'$ $\Psi \cup \Psi_Q \vdash c \prec a$.

Proof. Pick a name d that is fresh for all names under consideration. Using $\Psi \cup \Psi_P \vdash b \prec a$ we may infer $\Psi \cup \Psi_P \cup a \prec d \vdash b \curlyvee d$ and, using lemma `Semantics.comm1_aux` or `Semantics.comm2_aux` as the case might require, $\Psi \cup a \prec d \rhd P \xrightarrow[(\nu \widetilde{b}_P;\widetilde{z})b]{\underline{d}\, x} P'$. By simulation we have $\Psi \cup a \prec d \rhd [\![Q]\!] \xrightarrow[\pi]{\underline{d}\, x} Q'$ and $P' \mathrel{\dot\sim_{\Psi \cup a \prec d}} Q'$. By Theorem 1, $P' \mid (\nu d)(a \prec d) \mathrel{\dot\sim_\Psi} Q' \mid (\nu d)(a \prec d)$ and, because $(\nu d)(a \prec d) \mathrel{\dot\sim_{\mathbf{1}}} 0$, $P' \mathrel{\dot\sim_\Psi} Q'$. We may then use Lemmas 1 to find a sufficiently fresh provenance connected to d, and Lemma 8 to conclude that this provenance is below a.

Lemma 10. *If $\mathcal{F}([\![P]\!]) = (\nu \widetilde{b}_P)\Psi_P$ and $\Psi \cup \Psi_P \vdash \alpha \prec \gamma$ and $\widetilde{b}_P \# \Psi, \gamma$ then there exists β such that $\Psi \cup \Psi_P \vdash \beta \prec \gamma$ and $\Psi_P \vdash \alpha \prec \beta$ and $\widetilde{b}_P \# \beta$.*

Proof. By induction on the structure of P, using either Lemma 13 or 14 of [13] for each restriction depending on whether it binds into α or not.

Lemma 11. *If $\mathcal{F}([\![P]\!]) = (\nu \widetilde{b}_P)\Psi_P$ and $\mathcal{F}([\![Q]\!]) = (\nu \widetilde{b}_Q)\Psi_Q$ and $\Psi \cup \Psi_P \cup \Psi_Q \vdash a \curlyvee b$ and $\widetilde{b}_P \# \widetilde{b}_Q, \Psi_Q, \Psi, b$ and $\widetilde{b}_Q \# \Psi_Q, \Psi, a$ and Ψ is finitely supported with no $\curlyvee$ axioms, then there exists α, β such that $\widetilde{b}_P \# \alpha$ and $\widetilde{b}_Q \# \beta$ and $\Psi_P \vdash a \prec \alpha$ and $\Psi_Q \vdash b \prec \beta$ and $\Psi \cup \Psi_P \cup \Psi_Q \vdash \alpha \curlyvee \beta$.*

Proof. By case analysis.

- If $a \# \widetilde{b}_P$ and $b \# \widetilde{b}_Q$ we can choose $\alpha = a$ and $\beta = b$.
- If $a \# \widetilde{b}_P$ and $b \in \mathsf{n}(P)$ we can choose $\alpha = a$. Since $\Psi \cup \Psi_P \cup \Psi_Q$ contains only $\prec$ judgements, $\Psi \cup \Psi_P \cup \Psi_Q \vdash\vdash a \curlyvee b$ must be derivable via a chain of non-reflexive judgements $a_0 \prec a_1 \prec ... \prec a_n$ and $b_0 \prec b_1 \prec ... \prec b_m$ with

$a_n = b_m$ and $a_0 = a$ and $b_0 = b$, where each $a_i \prec a_{i+1}$ is an axiom of either Ψ, Ψ_P or Ψ_Q.

If there exists a least i such that $b_i \prec b_{i+1}$ and $b_{i+1}\#\widetilde{b}_P$, pick $\beta = b_{i+1}$. Otherwise, if there exists a greatest j such that $a_j \prec a_{j+1}$ and $a_j\#\widetilde{b}_P$, pick $\beta = \{a_j\}$. If no such i or j can be found we have a contradiction because $a_0 = a\#\widetilde{b}_P$.

– The remaining two cases are similar.

Lemma 12. *If $P \sim Q$ then $P[a := b] \sim Q[a := b]$*

Proof. Assume $a \neq b$; if not there is nothing to prove. We can prove that $(\nu a)(a/b \mid b/a \mid P) \sim P[a := b]$, in two steps.

First, we prove $\vdash a/b \mid b/a \mid P =\vdash a/b \mid b/a \mid P[a := b]$ by induction on the structure of P, using laws L11-L16 of [13, Table 2] to replace all free occurrences of b with a in the leftmost P.

Second, $\vdash (\nu a)(a/b \mid b/a \mid P) = (\nu a)(a/b \mid b/a \mid P[a := b]) = P[a := b]$ follows by scope extension and L23, and $(\nu a)(a/b \mid b/a \mid P) \sim P[a := b]$ by transitivity and soundness of the axiom scheme.

Thus by transitivity it suffices to prove $(\nu b)(a/b \mid b/a \mid P) \sim (\nu b)(a/b \mid b/a \mid Q)$, which follows immediately because $\sim$ is a congruence.

Lemma 13. *If $[\![P]\!] \dot{\sim}_{\mathbf{1}} [\![Q]\!]$ then $[\![P]\!][a := b] \dot{\sim}_{\mathbf{1}} [\![Q]\!][a := b]$.*

Proof. (Sketch) Along the same lines as Lemma 12, but instead of going via an axiomatisation we use the arcs a/b and b/a to mimic transitions by reconstructing connectivity judgements, and thus channel subjects. The idea is that any sequence of arcs in P that is broken by the substitution $P[a := b]$ because it goes via a can be reconstructed by going via b instead. Note that communication objects are not affected by the substitution because all outputs are bound in the range of the encoding.

Lemma 14. *If $\Psi \rhd P \xrightarrow{\underline{a}\,x} P'$ and $x\#a, P, \Psi$ then $\Psi \rhd P \xrightarrow{\underline{a}\,z} P'[x := z]$.*

Proof. A straightforward induction on the derivation of the transition.

Lemma 15 (Operational correspondence).

1. *If $P \xrightarrow{a(x)} P'$ and C is finitely supported then there is $P'', \widetilde{b}_P, \Psi_P, b$ such that $\mathbf{1} \rhd [\![P]\!] \xrightarrow[(\nu\widetilde{b}_P;\epsilon)b]{\underline{a}\,x} P''$ and $[\![P']\!] \dot{\sim}_{\mathbf{1}} P''$ and $\mathcal{F}([\![P]\!]) = (\nu\widetilde{b}_P)\Psi_P$ and $\widetilde{b}_P\#P, a, x, P', P'', C$ and $\Psi_P \vdash b \prec a$.*
2. *If $P \xrightarrow{\{a\}(x)} P'$ and C is finitely supported then there is $P'', \widetilde{b}_P, \Psi_P, b$ such that $\mathbf{1} \rhd [\![P]\!] \xrightarrow[(\nu\widetilde{b}_P;\epsilon)b]{\underline{a}\,x} P''$ and $[\![P']\!] \dot{\sim}_{\mathbf{1}} P''$ and $\mathcal{F}([\![P]\!]) = (\nu\widetilde{b}_P)\Psi_P$ and $\widetilde{b}_P\#P, a, x, P', P'', C$ and $\Psi_P \vdash b \curlyvee a$.*

3. *If* $P \xrightarrow{\overline{a}(x)} P'$ *and* C *is finitely supported then there is* $P'', \widetilde{b}_P, \Psi_P, b$ *such that* $\mathbf{1} \rhd [\![P]\!] \xrightarrow[(\nu \widetilde{b}_P;\epsilon)b]{\overline{a}(\nu x)x} P''$ *and* $[\![P']\!] \dot{\sim}_{\mathbf{1}} P''$ *and* $\mathcal{F}([\![P]\!]) = (\nu \widetilde{b}_P)\Psi_P$ *and* $\widetilde{b}_P \# P, a, C$ *and* $\Psi_P \vdash b \prec a$.
4. *If* $P \xrightarrow{\overline{\{a\}}(x)} P'$ *and* C *is finitely supported then there is* $P'', \widetilde{b}_P, \Psi_P, b$ *such that* $\mathbf{1} \rhd [\![P]\!] \xrightarrow[(\nu \widetilde{b}_P;\epsilon)b]{\overline{a}(\nu x)x} P''$ *and* $[\![P']\!] \dot{\sim}_{\mathbf{1}} P''$ *and* $\mathcal{F}([\![P]\!]) = (\nu \widetilde{b}_P)\Psi_P$ *and* $\widetilde{b}_P \# P, a, C$ *and* $\Psi_P \vdash b \curlyvee a$.
5. *If* $P \xrightarrow{[\varphi]\tau} P'$ *then there is* P'' *such that* $\varphi \rhd [\![P]\!] \xrightarrow{\tau} P''$ *and* $[\![P']\!] \mid [\![\varphi]\!] \dot{\sim}_{\mathbf{1}} P'' \mid [\![\varphi]\!]$.
6. *If* $\Psi \rhd [\![P]\!] \xrightarrow[(\nu \widetilde{b}_P;\widetilde{z})b]{\underline{a}\, x} P'$ *and* $y \# P, \Psi$ *and* $\mathcal{F}([\![P]\!]) = (\nu \widetilde{b}_P)\Psi_P$ *and* $\widetilde{b}_P, \widetilde{z} \# \Psi, P, a, \beta, y$ *and* $\Psi_P \vdash b \prec \beta$ *then there is* P'' *such that* $\mathsf{n}(\beta) \subseteq \mathsf{n}(P)$ *and* $P \xrightarrow{\beta(y)} P''$ *and* $[\![P'']\!][y := x] \dot{\sim}_{\Psi} P'$.
7. *If* $\Psi \rhd [\![P]\!] \xrightarrow[(\nu \widetilde{b}_P;\widetilde{z})b]{\overline{a}(\nu x)x} P'$ *and* $\mathcal{F}([\![P]\!]) = (\nu \widetilde{b}_P)\Psi_P$ *and* $\widetilde{b}_P, \widetilde{z} \# \Psi, P, a, \beta$ *and* $\Psi_P \vdash b \prec \beta$ *then there is* P'' *such that* $\mathsf{n}(\beta) \subseteq \mathsf{n}(P)$ *and* $P \xrightarrow{\overline{\beta}(y)} P''$ *and* $[\![P'']\!] \dot{\sim}_{\Psi} P'$.
8. *If* $\Psi \rhd [\![P]\!] \xrightarrow[\bot]{\tau} P'$ *then there exists* P'', φ *such that* $P \xrightarrow{[\varphi]\tau} P''$ *and* $\Psi \cup \mathcal{F}([\![P]\!]) \vdash \varphi$ *and* $[\![P'']\!] \dot{\sim}_{\Psi} P'$.

Proof. The clauses are proved in order and do not depend on subsequent clauses; hence the proofs of later clauses may use the earlier clauses.

1. By induction on the derivation of $P \xrightarrow{a(x)} P'$.

   In We have $a(y).P \xrightarrow{a(x)} (\nu y)(x/y \mid P)$ and $x \# a, y, P$. By rules In and Res (assuming y is chosen sufficiently fresh) we can derive

   $$\mathbf{1} \rhd (\nu y)(\underline{a}(\lambda x)x.((\!|y \prec x|\!) \mid [\![P]\!])) \xrightarrow[a]{\underline{a}\, x} (\nu y)((\!|y \prec x|\!) \mid [\![P]\!])$$

   The side-conditions are vacuous: frame freshness because there are no frame binders, bisimilarity of derivatives because they are syntactically equal, and the arc $\mathbf{1} \vdash a \prec a$ by rule $\vdash_{\text{Refl}}$.

   In-$\rhd$ We know that $P \xrightarrow{a(x)} P'$ and $P \rhd a \prec b$. By the inductive hypothesis there is $P'', \widetilde{b}_P, \Psi_P, c$ such that $\mathbf{1} \rhd [\![P]\!] \xrightarrow[(\nu \widetilde{b}_P;\epsilon)c]{\underline{a}\, x} P''$ and $[\![P']\!] \dot{\sim}_{\mathbf{1}} P''$ and $\mathcal{F}([\![P]\!]) = (\nu \widetilde{b}_P)\Psi_P$ and $\widetilde{b}_P \# P, a, x, P', P'', b, C$ and $\Psi_P \vdash c \prec a$. By Lemma 6 and definition of frame entailment, $\Psi_P \vdash a \prec b$. By rule $\vdash_{\text{Trans}}$ we have $\Psi_P \vdash c \prec b$, and by $\vdash_{\text{Join}}$ we have $\Psi_P \vdash c \curlyvee b$. By lemma `Semantics.comm2_aux` from the Isabelle formalisation we obtain

   $$\mathbf{1} \rhd [\![P]\!] \xrightarrow[(\nu \widetilde{b}_P;\epsilon)c]{\underline{b}\, x} P''$$

   Res By rule Res and Theorem 1.

Par-L By rule ParL and Theorem 1.
Par-R By rule ParR and Theorem 1.
Sum By rule Case and Theorem 1.

2. By induction on the derivation of $P \xrightarrow{\{a\}(x)} P'$. The only case where the proof differs from the case of unprotected inputs is for the In-$\triangleright$ rule. Here there are two subcases.
   - $P \xrightarrow{a(x)} P'$ and $P \rhd a \curlyvee b$. By Lemma 15.1 we obtain $P'', \widetilde{b}_P, \Psi_P, c$ such that $\mathbf{1} \rhd [\![P]\!] \xrightarrow[(\nu\widetilde{b}_P;\epsilon)c]{\underline{a}\,x} P''$ and $[\![P']\!] \dot{\sim}_{\mathbf{1}} P''$ and $\mathcal{F}([\![P]\!]) = (\nu\widetilde{b}_P)\Psi_P$ and $\widetilde{b}_P \# P, a, x, P', P'', b, C$ and $\Psi_P \vdash c \prec a$.
     By Lemma 6 and definition of frame entailment, $\Psi_P \vdash a \curlyvee b$. By rule $\vdash_{\text{Ext-Join}}$ we have $\Psi_P \vdash c \curlyvee b$. By lemma `Semantics.comm2_aux` from the Isabelle formalisation we obtain
     $$\mathbf{1} \rhd [\![P]\!] \xrightarrow[(\nu\widetilde{b}_P;\epsilon)c]{\underline{b}\,x} P''$$
   - $P \xrightarrow{\{a\}(x)} P'$ and $P \rhd b \prec a$. By the induction hypothesis we obtain $P'', \widetilde{b}_P, \Psi_P, c$ such that $\mathbf{1} \rhd [\![P]\!] \xrightarrow[(\nu\widetilde{b}_P;\epsilon)c]{\underline{a}\,x} P''$ and $[\![P']\!] \dot{\sim}_{\mathbf{1}} P''$ and $\mathcal{F}([\![P]\!]) = (\nu\widetilde{b}_P)\Psi_P$ and $\widetilde{b}_P \# P, a, x, P', P'', b, C$ and $\Psi_P \vdash c \curlyvee a$.
     By Lemma 6 and definition of frame entailment, $\Psi_P \vdash b \prec a$. By rules $\vdash_{\text{Ext-Join}}$ and $\vdash_{\text{Mirror}}$ we have $\Psi_P \vdash c \curlyvee b$. By lemma `Semantics.comm2_aux` from the Isabelle formalisation we obtain
     $$\mathbf{1} \rhd [\![P]\!] \xrightarrow[(\nu\widetilde{b}_P;\epsilon)c]{\underline{b}\,x} P''$$
3. By induction on the derivation of $P \xrightarrow{\overline{a}(x)} P'$. The proofs are the same as for the input case.
4. By induction on the derivation of $P \xrightarrow{\overline{\{a\}}(x)} P'$. The proofs are the same as for the input case.
5. By induction on the derivation of $P \xrightarrow{[\varphi]\tau} P'$.
   Tau Here $[\varphi]\tau.P \xrightarrow{[\varphi]\tau} P'$ Using rules In, Out, Com, Res and Case we can derive $\varphi \rhd \mathbf{case}\ \varphi : (\nu x)(\underline{x}(\lambda x)x.0 \mid \overline{x}x.[\![P]\!]) \xrightarrow[\perp]{\tau} (\nu x)(0 \mid [\![P]\!])$. $[\![P]\!] \mid [\![\varphi]\!] \dot{\sim}_{\mathbf{1}} (\nu x)(0 \mid [\![P]\!]) \mid [\![\varphi]\!]$ follows by Theorem 2 (using $x \# P$ and hence $x \# [\![P]\!]$ to push the restriction down to 0).
   Comm-L We know that $P \xrightarrow{\overline{a}(x)} P'$ and $Q \xrightarrow{\overline{b}(x)} Q'$. By the first two clauses of Lemma 15 we obtain $\widetilde{b}_P, \Psi_P, \widetilde{b}_Q, \Psi_Q, c, d, P'', Q''$ such that $\mathbf{1} \rhd [\![P]\!] \xrightarrow[(\nu\widetilde{b}_P;\epsilon)c]{\underline{a}\,x} P''$ and $[\![P']\!] \dot{\sim}_{\mathbf{1}} P''$ and $\mathcal{F}([\![P]\!]) = (\nu\widetilde{b}_P)\Psi_P$ and $\widetilde{b}_P \# P, Q, Q', a, b, x, P', P'', \widetilde{b}_Q, \Psi_Q$ and $\Psi_P \vdash c \prec a$ and $\mathbf{1} \rhd [\![Q]\!] \xrightarrow[(\nu\widetilde{b}_Q;\epsilon)d]{\overline{b}(\nu x)x} Q''$ and $[\![Q']\!] \dot{\sim}_{\mathbf{1}} Q''$ and $\mathcal{F}([\![Q]\!]) = (\nu\widetilde{b}_Q)\Psi_Q$ and $\widetilde{b}_Q \# Q, a, b, C, P, P'', P'', \Psi_P$ and $\Psi_Q \vdash d \prec b$.

By monotonicity of $\vdash$ and lemma `Semantics.transfer_frame` from the formalisation, we can strengthen the frames of the transitions to

$$\Psi_Q \cup \Psi \rhd [\![P]\!] \xrightarrow[(\nu \widetilde{b}_P;\epsilon)c]{\underline{a}\, x} P''$$

and

$$\Psi_P \cup \Psi \rhd [\![Q]\!] \xrightarrow[(\nu \widetilde{b}_Q;\epsilon)d]{\overline{b}(\nu x)x} Q''$$

where $\Psi = \{a \prec e, b \prec e\}$ for some e fresh for all names under consideration. By monotonicity of $\vdash$ we have that $\Psi_P \cup \Psi_Q \cup \Psi \vdash c \prec a$ and $\Psi_P \cup \Psi_Q \cup \Psi \vdash d \prec b$. By $\vdash_{ExtJoin}$ and $\vdash_{Mirror}$ we then have that the provenances are joinable:

$$\Psi_P \cup \Psi_Q \cup \Psi \vdash c \curlyvee d$$

This lets us use lemmas `Semantics.comm1_aux` and `Semantics.comm2_aux` from the formalisation to relabel the transitions as follows:

$$\Psi_Q \cup \Psi \rhd [\![P]\!] \xrightarrow[(\nu \widetilde{b}_P;\epsilon)c]{\underline{d}\, x} P''$$

and

$$\Psi_P \cup \Psi \rhd [\![Q]\!] \xrightarrow[(\nu \widetilde{b}_Q;\epsilon)d]{\overline{b}(\nu x)x} Q''$$

By rule Com we can derive

$$\Psi \rhd [\![P]\!] \mid [\![Q]\!] \xrightarrow[\perp]{\tau} (\nu x)(P'' \mid Q'')$$

Since $(\nu e)\Psi \mathrel{\dot{\sim}_{\mathbf{1}}} a \curlyvee b$, All that remains is to show that $(\nu x)(P'' \mid Q'') \mathrel{\dot{\sim}_{\Psi}} (\nu x)([\![P']\!] \mid [\![Q']\!])$, which follows from $P'' \mathrel{\dot{\sim}_{\mathbf{1}}} [\![P']\!]$ and $Q'' \mathrel{\dot{\sim}_{\mathbf{1}}} [\![Q']\!]$ by Theorems 1–2 and extension of arbitrary assertion.

Comm-R This case is symmetric to Comm-L.

Tau-$\rhd$ Follows from lemma `Semantics.transfer_frame` from the Isabelle formalisation.

Res By rule Res and Theorem 1.

Par-L By rule ParL and Theorem 1.

Par-R By rule ParR and Theorem 1.

6. By structural induction on P, followed by inversion on the derivation of the transition from $[\![P]\!]$.

   $P = a(x).Q$ We know that $\Psi \rhd [\![P]\!] \xrightarrow[a]{\underline{b}\, y} (\nu x)((\!| x \prec y |\!) \mid [\![Q]\!])$. Because $\Psi_P = \{\}$ it must be the case that either $\beta = a$ or $\beta = \{a\}$. In either case, we may choose a fresh name z and derive—by rule In and, if β is protected, In-$\rhd$—the following:

   $$P \xrightarrow{\beta(z)} (\nu x)(z/x \mid Q)$$

   The remaining conjuncts are easily discharged: support inclusion by $\mathsf{n}(\beta) = \{a\} \subseteq \mathsf{n}(P)$, bisimilarity because of reflexivity and freshness of y.

$P = (\nu z)Q$ The transition from $[\![P]\!]$ must have been derived by the Res rule. We thus know that $\Psi \rhd Q \xrightarrow[(\nu \widetilde{b}_Q;\widetilde{z})b]{\underline{a}\,x} Q'$ where $z\#\Psi, a, x, \beta, y$ and $y\#P, \Psi$ and $\mathcal{F}([\![Q]\!]) = (\nu\widetilde{b}_Q)\Psi_Q$ and $\widetilde{b}_Q, \widetilde{z}, z\#\Psi, P, \beta, y$ and $\Psi_Q \vdash b \prec \beta$. Using $y\#z$ we get $y\#Q$.

By applying the induction hypothesis we obtain Q'' such that $Q \xrightarrow{\beta(y)} Q''$ and $\mathsf{n}(\beta) \subseteq \mathsf{n}(Q)$ and $[\![Q'']\!][y := x] \mathrel{\dot{\sim}}_\Psi Q'$. Because $z\#\beta, y$ we can derive $(\nu z)Q \xrightarrow{\beta(y)} (\nu z)Q''$. By Theorem 1, $(\nu z)([\![Q'']\!][y := x]) \mathrel{\dot{\sim}}_\Psi Q'$. To conclude it suffices to note that $(\nu z)([\![Q'']\!][y := x]) = ((\nu z)[\![Q'']\!])[y := x]$ because $y, x\#z$.

$P = Q \mid R$ We consider only the case where the transition was derived via tha Par-L rule (the other case is symmetric). In other words, we have that
$\Psi \cup \Psi_R \rhd [\![Q]\!] \xrightarrow[(\nu\widetilde{b}_Q;\widetilde{z})b]{\underline{a}\,x} Q'$ $y\#P, Q, \Psi$ and $\mathcal{F}([\![Q]\!]) = (\nu\widetilde{b}_Q)\Psi_Q$ and $\mathcal{F}([\![R]\!]) = (\nu\widetilde{b}_R)\Psi_R$ and $\widetilde{b}_Q, \widetilde{b}_R, \widetilde{z}\#\Psi, Q, R, \beta, y$ and $\widetilde{b}_R\#\widetilde{b}_Q, b, \widetilde{z}, \Psi_Q$ and $\Psi_Q \cup \Psi_R \vdash b \prec \beta$.

By Lemma 10 there is γ such that $\Psi_Q \cup b \prec \gamma$ and $\Psi_Q \cup \Psi_R\gamma \prec \beta$ and $\Psi_Q\#\gamma$.

By the induction hypothesis there is Q'' such that $Q \xrightarrow{\gamma(y)} Q''$ and $[\![Q'']\!][y := x] \mathrel{\dot{\sim}}_\Psi Q'$ and $\mathsf{n}(\gamma) \subseteq \mathsf{n}(Q)$. The latter yields $\widetilde{b}_R\#\gamma$, which suffices to derive $P \mid R \rhd \gamma \prec \beta$. By rules Par-L and In-$\rhd$, $Q \mid R \xrightarrow{\beta(y)} Q'' \mid R$

Because $y\#R$, Theorem 1 yields $[\![Q'' \mid R]\!][y := x] \mathrel{\dot{\sim}}_\Psi Q' \mid R$.

$P = \Sigma_i.\pi_i.P_i$ Similar to the case for input, using also rule Case to derive the matching transition.

7. Analogous to the input case, but easier because we only deal in bound outputs; hence no need to consider a substitution.
8. By structural induction P, followed by inversion on the derivation of $\Psi \rhd [\![P]\!] \xrightarrow[\bot]{\tau} P'$. The interesting case is when $P = Q \mid R$ and the transition from $[\![P]\!]$ is derived via the Com rule. There we have that $\Psi\cup\Psi_R \rhd [\![Q]\!] \xrightarrow[(\nu\widetilde{b}_Q;\widetilde{z})b]{\overline{a}(\nu x)x} Q'$ and $\Psi\cup\Psi_Q \rhd [\![R]\!] \xrightarrow[(\nu\widetilde{b}_R;\widetilde{y})b]{\underline{b}\,x} R'$ and $\mathcal{F}([\![Q]\!]) = (\nu\widetilde{b}_Q)\Psi_Q$ and $\mathcal{F}([\![R]\!]) = (\nu\widetilde{b}_R)\Psi_R$ and $\widetilde{b}_Q\#\Psi, Q, R, a, \widetilde{z}, \widetilde{y}, \widetilde{b}_R, \Psi_R$ and $\widetilde{b}_R\#\Psi, Q, R, b, \widetilde{z}, \widetilde{b}_R, \Psi_Q$ and $x\#\Psi, R$ and $\widetilde{z}, \widetilde{y}$ are similarly fresh.
By Lemma 1 $\Psi\cup\Psi_Q\cup\Psi_R a \curlyvee b$. By Lemma 11 we get α, β such that $\widetilde{b}_Q \vdash b \prec \alpha$ and $\widetilde{b}_R \vdash a \prec \beta$ and $\widetilde{b}_Q\#\alpha$ and $\widetilde{b}_R\#\beta$ and $\Psi \cup \Psi_R \cup \Psi_Q \vdash \alpha \curlyvee \beta$.
By Lemmas 15.6 and 15.7 (instantiated with the identity substitution) we can derive Q'', R'' such that $Q \xrightarrow{\overline{\alpha}(x)} Q''$ and $R \xrightarrow{\overline{\beta}(x)} R''$ and $[\![Q'']\!] \mathrel{\dot{\sim}}_{\Psi\cup\Psi_R} Q'$ and $[\![R'']\!] \mathrel{\dot{\sim}}_{\Psi\cup\Psi_Q} R'$ and $\mathsf{n}(\alpha) \subseteq \mathsf{n}(Q)$ and $\mathsf{n}(\beta) \subseteq \mathsf{n}(R)$. By Theorem 1 and Theorem 2, $[\![(\nu x)(Q'' \mid R'')]\!] \mathrel{\dot{\sim}}_\Psi (\nu x)(Q' \mid R')$.

By Comm-L, $Q \mid R \xrightarrow{[\alpha \curlyvee \beta]\tau} (\nu x)(Q'' \mid R'')$. It suffices to show that $\Psi \cup \mathcal{F}(P) \vdash \alpha \curlyvee \beta$; this follows from $\Psi \cup \Psi_R \cup \Psi_Q \vdash \alpha \curlyvee \beta$ and the fact that α, β are fresh wrt. the frame binders.

Lemma 16. *If $P \sim Q$ then $[\![P]\!] \mathrel{\dot{\sim}_{\mathbf{1}}} [\![Q]\!]$.*

Proof. By showing that $\mathcal{R} = \{(P, Q).[\![P]\!] \mathrel{\dot{\sim}_{\mathbf{1}}} [\![Q]\!]\}$ is a bisimulation relation. We consider each kind of transition label separately, ignoring the cases for output (which are similar to, but easier than, input):

- We know that $P \rhd \varphi$. By Lemma 6 we obtain $\mathcal{F}([\![P]\!]) \vdash \varphi$. By static equivalence $\mathcal{F}([\![Q]\!]) \vdash \varphi$, and by Lemma 6 we conclude $Q \rhd \varphi$.
- We have $P \xrightarrow{a(x)} P'$ and $[\![P]\!] \mathrel{\dot{\sim}_{\mathbf{1}}} [\![Q]\!]$. By Lemma 15 there is $P'', \widetilde{b}_P, \Psi_P, b$ such that $\mathbf{1} \rhd [\![P]\!] \xrightarrow[(\nu \widetilde{b}_P; \epsilon) b]{\underline{a}\, x} P''$ and $[\![P']\!] \mathrel{\dot{\sim}_{\mathbf{1}}} P''$ and $\mathcal{F}([\![P]\!]) = (\nu \widetilde{b}_P)\Psi_P$ and $\widetilde{b}_P \# P, a, x, P', P'', Q$ and $\Psi_P \vdash b \prec a$.
  By Lemma 9 (choosing suitably fresh frame binders for $[\![Q]\!]$) we obtain $\mathbf{1} \rhd Q \xrightarrow[(\nu \widetilde{b}_Q; \widetilde{y}) c]{\underline{a}\, x} Q''$ such that $\Psi_Q \vdash c \prec a$ and $P'' \mathrel{\dot{\sim}_{\mathbf{1}}} Q''$.
  By Lemma 15.6 (choosing $y = x$) we conclude that there is $Q \xrightarrow{a} Q'$ such that $[\![Q']\!] \mathrel{\dot{\sim}_{\mathbf{1}}} Q''[x := x] = Q''$. that $[\![P']\!] \mathrel{\dot{\sim}_{\mathbf{1}}} [\![Q']\!]$ follows by associativity and commutativity.
- We have $P \xrightarrow{\{a\}(x)} P'$ and $[\![P]\!] \mathrel{\dot{\sim}_{\mathbf{1}}} [\![Q]\!]$. By Lemma 15 there is $P'', \widetilde{b}_P, \Psi_P, b$ such that $\mathbf{1} \rhd [\![P]\!] \xrightarrow[(\nu \widetilde{b}_P; \epsilon) b]{\underline{a}\, x} P''$ and $[\![P']\!] \mathrel{\dot{\sim}_{\mathbf{1}}} P''$ and $\mathcal{F}([\![P]\!]) = (\nu \widetilde{b}_P)\Psi_P$ and $\widetilde{b}_P \# P, a, x, P', P'', Q$ and $\Psi_P \vdash b \curlyvee a$.
  By simulation and Lemma 1, we obtain Q'', c such that $\mathbf{1} \rhd [\![Q]\!] \xrightarrow[(\nu \widetilde{b}_Q; \widetilde{z}) c]{\underline{a}\, x} Q''$. and $\Psi_Q \vdash a \curlyvee c = c \prec \{a\}$ for some suitable fresh frame binder $\widetilde{b}_Q$. By Lemma 15.6 (choosing $y = x$) we conclude that there is $Q \xrightarrow{\{a\}} Q'$ such that $[\![Q']\!] \mathrel{\dot{\sim}_{\mathbf{1}}} Q''[x := x] = Q''$. That $[\![P']\!] \mathrel{\dot{\sim}_{\mathbf{1}}} [\![Q']\!]$ follows by associativity and commutativity.
- We know that $P \xrightarrow{[\varphi]\tau} P'$ and $[\![P]\!] \mathrel{\dot{\sim}_{\mathbf{1}}} [\![Q]\!]$. By Lemma 15 there is P'' such that $\varphi \rhd [\![P]\!] \xrightarrow{\tau} P''$ and $[\![P']\!] \mid [\![\varphi]\!] \mathrel{\dot{\sim}_{\mathbf{1}}} P'' \mid [\![\varphi]\!]$. By extension of arbitrary assertion and simulation, there is Q'' such that $\varphi \rhd [\![Q]\!] \xrightarrow{\tau} Q''$ and $P'' \mathrel{\dot{\sim}_{\varphi}} Q''$. From Lemma 15 we obtain Q', φ' such that $[\![Q']\!] \mathrel{\dot{\sim}_{\varphi}} Q''$ and $Q \xrightarrow{[\varphi']\phi} Q'$ and $\varphi \cup \mathcal{F}(P) \vdash \varphi'$ By Lemma 6 and rule Tau-$\rhd$ we obtain $Q \xrightarrow{[\varphi]\phi} Q'$. All that then remains is to show that $[\![P \mid \varphi]\!] \mathrel{\dot{\sim}_{\mathbf{1}}} [\![Q \mid \varphi]\!]$, which follows by Theorem 1 and the definition of $[\![\_]\!]$.

Lemma 17. *If $[\![P]\!] \mathrel{\dot{\sim}_{\mathbf{1}}} [\![Q]\!]$ then $P \sim Q$.*

Proof. By showing that $\mathcal{R} = \{(\Psi, [\![P]\!], [\![Q]\!]).P \mid \Psi \sim Q \mid \Psi\}$ is a bisimulation up to $\mathrel{\dot{\sim}}$. There are four cases to consider:

- Static equivalence is immediate from Lemma 6.
- Symmetry is immediate from the symmetry of $\sim$.
- Extension of arbitrary assertion is immediate from the compositionality and associativity of $\sim$.
- Simulation. We know that $\Psi \rhd [\![P]\!] \xrightarrow[\pi]{\alpha} P'$ and $\mathrm{bn}(\alpha)\#P$ and $P \mid \Psi \sim Q \mid \Psi$, and proceed by case analysis on α.
  - $\alpha = \underline{a}\, x$. We use Lemma 1 to obtain $\widetilde{b}_P, \widetilde{b}_Q, \widetilde{x}, b$ such that $\mathcal{F}([\![P]\!]) = (\nu\widetilde{b}_P)\Psi_P$ and $\pi = (\nu\widetilde{b}_P;\widetilde{x})b$ and $\widetilde{b}_P\#\Psi, P, a, x, P', Q, \widetilde{x}$ and $\widetilde{x}\#\Psi, P, a, P', Q$ and $\Psi \cup \Psi_P \vdash a \curlyvee b$. Since $a \curlyvee b = b \prec \{a\}$ we can use Lemma 10 to obtain a β such that $\Psi_P \vdash b \prec \beta$ and $\Psi \cup \Psi_P \vdash \beta \prec \{a\}$ and $\beta\#\widetilde{b}_P$.
    Choosing a fresh name $y\#\Psi, P, Q, a, x, P'$ and using Lemma 15.6 we obtain P'' such that $P \xrightarrow{\beta(y)} P''$ and $[\![P'']\!][y := x] \mathrel{\dot\sim}_\Psi P'$. By simulation, and inversion on the derivation of the resulting transition from $Q \mid \Psi$, we get Q'' such that $Q \mid \Psi \xrightarrow{\beta} Q'' \mid \Psi$ and $P'' \mid \Psi \sim Q'' \mid \Psi$. By static equivalence, freshness of β, a and Lemma 6 we get $Q \mid \Psi \rhd \beta \prec \{a\}$, and by rule In-$\rhd$, $Q \mid \Psi \xrightarrow{\{a\}y} Q'' \mid \Psi$. Lemma 15.2, and inversion on the derivation of the resulting transition from $[\![Q]\!] \mid \Psi$, yields a Q' such that $\mathbf{1} \rhd [\![Q]\!] \mid [\![\Psi]\!] \xrightarrow{\underline{a}\, y} Q' \mid [\![\Psi]\!]$. and $Q' \mid [\![\Psi]\!] \mathrel{\dot\sim}_{\mathbf{1}} [\![Q'']\!] \mid [\![\Psi]\!]$. By Lemmas 13 and 14, $\mathbf{1} \rhd [\![Q]\!] \mid [\![\Psi]\!] \xrightarrow{\underline{a}\, x} Q'[y := x] \mid [\![\Psi]\!]$ and $Q'[y := x] \mid [\![\Psi]\!] \mathrel{\dot\sim}_{\mathbf{1}} [\![Q'']\!][y := x] \mid [\![\Psi]\!]$, which implies $\Psi \rhd [\![Q]\!] \xrightarrow{\underline{a}\, x} Q'[y := x]$ and $Q'[y := x] \mathrel{\dot\sim}_\Psi [\![Q'']\!][y := x] = [\![Q''[y := x]]\!]$. By Lemma 12, $P''[y := x] \mid \Psi \sim Q''[y := x] \mid \Psi$.
    We now have all we need to conclude
    $$P' \mathrel{\dot\sim}_\Psi [\![P''[y := x]]\!]\ \mathcal{R}_\Psi\ [\![Q''[y := x]]\!] \mathrel{\dot\sim}_\Psi Q'[y := x]$$
    which closes the bisimulation diagram up to $\dot\sim$.
  - $\alpha = \overline{a}(\nu x)x$. Elided. Analogous to the input case, but easier because there is no need to consider substitutions.
  - $\alpha = \tau$ (and hence $\pi = \bot$). By Lemma 15.8 there is P'', φ such that $\Psi \cup \mathcal{F}(P) \vdash \varphi$ and $P \xrightarrow{[\varphi]\tau} P''$ and $[\![P'']\!] \mathrel{\dot\sim}_\Psi P'$. By simulation there is Q'' such that $Q \xrightarrow{[\varphi]\tau} Q''$ and $P' \mathrel{\dot\sim}_\Psi Q''$.
    By static equivalence and Lemma 6, and by rules Par-R and Tau-$\rhd$, $Q \mid \Psi \xrightarrow{\tau} Q'' \mid \Psi$. Then $[\![Q]\!]$ can mimic the transition by Lemma 15.5 (instantiated with $\varphi = y \prec y$ for some fresh y), yielding a Q' such that $a \prec a \rhd [\![Q]\!] \mid [\![\Psi]\!] \xrightarrow{\tau} Q' \mid [\![\Psi]\!]$ and $[\![Q'']\!] \mid [\![\Psi]\!] \mid (\!|y \prec y|\!) \mathrel{\dot\sim}_{\mathbf{1}} Q' \mid [\![\Psi]\!] \mid (\!|y \prec y|\!)$. Using Lemma 7 this is equivalent to $[\![Q'']\!] \mathrel{\dot\sim}_\Psi Q'$
    Thus we have established $P'' \mathrel{\dot\sim}_\Psi [\![P'']\!]\ \mathcal{R}\ [\![Q'']\!] \mathrel{\dot\sim}_\Psi Q''$ which closes the bisimulation diagram up to $\dot\sim$.

Mixed choice

Definition 17. *Let $\mathcal{P} = (\mathbf{T}, \mathbf{A}, \mathbf{C}, \vdash, \otimes, \mathbf{1}, \dot\to)$ be a psi-calculus. Then $\mathcal{E}(\mathcal{P}) = (\mathbf{T}_\mathcal{E}, \mathbf{A}_\mathcal{E}, \mathbf{C}_\mathcal{E}, \vdash_\mathcal{E}, \otimes_\mathcal{E}, \mathbf{1}_\mathcal{E}, \dot\to_\mathcal{E})$ is a psi-calculus whose components are as follows:*

$$\mathbf{T}_{\mathcal{E}} = \mathbf{T} \uplus \{M_x : x \in \mathcal{N}, M \in \mathbf{T}_{\mathcal{E}}\} \qquad \mathbf{A}_{\mathcal{E}} = \mathbf{A} \times \mathcal{P}_{\mathit{fin}}(\mathcal{N})$$

$$\mathbf{C}_{\mathcal{E}} = \mathbf{C} \uplus \{M \dot{\to} N : M, N \in \mathbf{T}\} \uplus \mathcal{N} \qquad (\Psi, \mathbf{N}) \otimes_{\mathcal{E}} (\Psi', \mathbf{N}') = (\Psi \otimes \Psi', \mathbf{N} \cup \mathbf{N}')$$

$$\mathbf{1}_{\mathcal{E}} = (\mathbf{1}, \emptyset) \qquad (\Psi, \mathbf{N}) \vdash_{\mathcal{E}} \varphi \quad \textit{if } \varphi \in \mathbf{C} \textit{ and } \Psi \vdash \varphi$$

$$(\Psi, \mathbf{N}) \vdash_{\mathcal{E}} x \quad \textit{if } x \in \mathcal{N} \textit{ and } x \in \mathbf{N}$$

$$(\Psi, \mathbf{N}) \vdash_{\mathcal{E}} M_x \dot{\to} N_y \quad \textit{if } \Psi \vdash M \dot{\to} N \textit{ and } x \neq y \textit{ and } x, y \notin \mathbf{N}$$

$$(\Psi, \mathbf{N}) \vdash_{\mathcal{E}} M_x \dot{\to} N \quad \textit{if } \Psi \vdash M \dot{\to} N \textit{ and } x \notin \mathbf{N}$$

$$(\Psi, \mathbf{N}) \vdash_{\mathcal{E}} M \dot{\to} N_x \quad \textit{if } \Psi \vdash M \dot{\to} N \textit{ and } x \notin \mathbf{N}$$

We will sometimes write $\mathbf{N}$ to abbreviate $(\mathbf{1}, \mathbf{N})$ or $(\!|(\mathbf{1}, \mathbf{N})|\!)$.

Before proceeding we will make—and justify—a few simplifying assumptions for presentation purposes.

First, we assume that in the source language under consideration $\mathbf{1}\sigma = \mathbf{1}$ for all substitutions σ. This is to ensure that input-guarded assertions $(\mathbf{1}, \{x\})$, whose sole purpose is to disable the tag x, has the same effect on the source-language environment as if there had been no assertion at all underneath the input prefix. Otherwise, the encoding fails because the target term may expose the assertion $\mathbf{1}\sigma$ even if the source term does not, thus potentially entailing different conditions.

This assumption can be easily lifted by allowing target-language assertions to be taken from $\mathcal{P}_{\text{fin}}(\mathcal{N})$, ie. with no source-language assertion attached to it, and using such assertions underneath prefix instead. We have opted against this presentation style in order to avoid redundancy in the definition of assertion composition.

Second, we assume that the target language is constrained by a sorting system as in [4] that ensures only terms $M \in \mathbf{T}$ can ever occur as objects of communication, and in particular, that for all substitutions $[\widetilde{x} := \widetilde{T}]$, $\widetilde{T} \subseteq \mathbf{T}$. The purpose of this simplification is to avoid having to consider input transitions such as

$$\Psi \rhd \underline{M}(\lambda \widetilde{x}) N.P \xrightarrow{\underline{K}\, K_x} P'$$

that may result in substitutions where tagged terms must be substituted into source-language terms or vice versa.

This assumption can be lifted by changing the target-language terms as follows:

$$\begin{aligned} \mathbf{T}_{\mathcal{E}} := \; & \mathbf{T} \\ | \; & \mathbf{T}_{\mathcal{E}}(\mathcal{N}) \\ | \; & \mathbf{T}_{\mathcal{E}}\langle \mathcal{N} \rangle \end{aligned}$$

where terms $M(x)$ play the same role as terms M_x, and terms $M\langle x \rangle$ serve as input patterns to prevent receipt of tagged terms. Let $M\langle \widetilde{y} \rangle$ abbreviate $M\langle y_0 \rangle \langle y_1 \rangle \ldots \langle y_n \rangle$. Specifically, we would encode input as

$$[\![\underline{M}(\lambda\widetilde{x})N.P]\!] = (\nu y)\underline{M}(\lambda\widetilde{x})N\langle\widetilde{x},y\rangle.[\![P]\!]$$

where y is a fresh name. Substitution on terms $N\langle y\rangle$ is then defined as

$$\begin{aligned}
N\langle\widetilde{z},y\rangle[\widetilde{x}:=\widetilde{T}] &= N[\widetilde{x}:=\widetilde{T}] \text{if } y\#\widetilde{x},\widetilde{T} \text{ and } \widetilde{z}=\widetilde{x} \text{ and } \widetilde{T}\subseteq\mathbf{T}\\
N\langle\widetilde{z},y\rangle[\widetilde{x}:=\widetilde{T}] &= N[\widetilde{x}:=\widetilde{T}]\langle\widetilde{z},y\rangle \text{if } y\#\widetilde{x},\widetilde{T} \text{ and } \widetilde{z}\#\widetilde{x},\widetilde{T} \text{ and } \widetilde{T}\subseteq\mathbf{T}\\
&= T_o\langle\widetilde{w}\rangle \text{where } \widetilde{w}=\mathsf{n}(\widetilde{T})\cup((\mathsf{n}(N)\cup\widetilde{x}\cup y)-\widetilde{x}) \text{ otherwise}
\end{aligned}$$

The idea is that because we maintain the invariant that y is (ν)-bound directly above all input prefixes, any message M matching pattern $(\lambda\widetilde{x})N\langle\widetilde{x},y\rangle$ via a substitution $N\langle\widetilde{x},y\rangle[\widetilde{x}:=\widetilde{T}]=M$, all of $\widetilde{T}$ are source-language terms and $M=N[\widetilde{x}:=\widetilde{T}]$ (by Clause 1); hence M is a source-language term because the set of source-language terms are closed under substitution. If any of $\widetilde{T}$ are target-language terms, Clause 3 of the definition of substitution applies, and we have $y\in\mathsf{n}(M)$ which contradicts freshness of y. The second clause ensures that for nested input prefixes, the tag y is not removed by subtitutions applied to the inner prefixes as a result of consuming the outer prefixes. The somewhat convoluted definition of Clause 3 is to enforce the requisite that all names substituted into the term are present in the result of substitution.

Lemma 18. $(\Psi,\mathbf{N})\simeq(\Psi',\mathbf{N}')$ *iff* $\Psi\simeq\Psi'$ *and* $\mathbf{N}=\mathbf{N}'$

Proof. Easy since entailment $\vdash_{\mathcal{E}}$ is only defined in terms of membership in $\mathbf{N}$ and entailment on Ψ.

Lemma 19. *If $\mathcal{P}$ is a valid psi-calculus then so is $\mathcal{E}(\mathcal{P})$*

Proof. Associativity and commutativity of $\otimes_{\mathcal{E}}$ are immediate from the corresponding properties of $\otimes$ and $\cup$. Compositionality follows from Lemma 18 and the compositionality of $\otimes$ (up to $\simeq$) and $\cup$ (up to extensional equality).

Definition 18 (Encoding). *The encoding $[\![\_]\!]$ from processes of $\mathcal{P}$ to processes of $\mathcal{E}(\mathcal{P})$ is homomorphic on all operators except choice and assertion, where it is defined as follows:*

$$[\![(\!|\Psi|\!)]\!] = (\!|(\Psi,\emptyset)|\!) \qquad [\![\alpha.P+\beta.Q]\!] = (\nu x)(\alpha.(P\mid(\!|(\mathbf{1},\{x\})|\!)\mid\beta.(Q\mid(\!|(\mathbf{1},\{x\})|\!)$$

where $x\#\alpha,\beta,P,Q$

Lemma 20. *If $\widetilde{T}\subseteq\mathbf{T}$ then $[\![P]\!][\widetilde{x}:=\widetilde{T}]=[\![P[\widetilde{x}:=\widetilde{T}]]\!]$*

Proof. By a straightforward structural induction on P.

Lemma 21. $(\nu x)((\!|(\mathbf{1},\{x\})|\!)\mid\alpha_x.P)\mathrel{\dot{\sim}}_{\Psi}0$

Proof. (Sketch) Static equivalence follows because the $(\mathbf{1},\{x\})$ affects only judgements involving the name x, which are not considered outside the scope of the ν binder. Simulation follows because since α_x is disabled, no transitions can be derived.

Lemma 22. *$\mathcal{F}([\![P]\!]) = (\nu \widetilde{b}_P, y)(\Psi_P, \emptyset)$, where $\mathcal{F}(P) = (\nu \widetilde{b}_P)\Psi_P$ and $y \# P, \Psi_P, \widetilde{b}_P$.*

Proof. A straightforward structural induction on P.

Theorem 8 (Operational correspondence).

1. *If $\Psi \rhd P \xrightarrow{\underline{M}\, N} P'$ then there is P'' such that $(\Psi, \emptyset) \rhd [\![P]\!] \xrightarrow{\underline{M}\, N} P''$ and $P'' \mathrel{\dot\sim}_{(\Psi,\emptyset)} [\![P']\!]$.*
2. *If $\Psi \rhd P \xrightarrow{\overline{M}(\widetilde{x})N} P'$ then there is P'' such that $(\Psi, \emptyset) \rhd [\![P]\!] \xrightarrow{\overline{M}(\widetilde{x})N} P''$ and $P'' \mathrel{\dot\sim}_{(\Psi,\emptyset)} [\![P']\!]$.*
3. *If $\Psi \rhd P \xrightarrow{\tau} P'$ then there is P'' such that $(\Psi, \emptyset) \rhd [\![P]\!] \xrightarrow{\tau} P''$ and $P'' \mathrel{\dot\sim}_{(\Psi,\emptyset)} [\![P']\!]$.*
4. *If $(\Psi, \mathbf{N}) \rhd [\![P]\!] \xrightarrow{\underline{M}\, N} P'$ then there is P'' such that $\Psi \rhd P \xrightarrow{\underline{M_\perp}\, N} P''$ and $P' \mathrel{\dot\sim}_{(\Psi,\mathbf{N})} [\![P'']\!]$.*
5. *If $(\Psi, \mathbf{N}) \rhd [\![P]\!] \xrightarrow{\overline{M}(\nu \widetilde{x})N} P'$ then there is P'' such that $\Psi \rhd P \xrightarrow{\overline{M_\perp}(\nu \widetilde{x})N} P''$ and $P' \mathrel{\dot\sim}_{(\Psi,\mathbf{N})} [\![P'']\!]$.*
6. *If $(\Psi, \mathbf{N}) \rhd [\![P]\!] \xrightarrow{\tau} P'$ then there is P'' such that $\Psi \rhd P \xrightarrow{\tau} P''$ and $P' \mathrel{\dot\sim}_{(\Psi,\mathbf{N})} [\![P'']\!]$.*

Proof. 1. By induction on the derivation of the transition from P. The interesting case is choice. There we have $\Psi \rhd \underline{M}(\lambda \widetilde{x})N.P + \alpha.Q \xrightarrow[M]{\underline{K}\, N[\widetilde{x} := \widetilde{T}]} P[\widetilde{x} := \widetilde{T}]$ and $\Psi \vdash K \mathrel{\dot\to} M$. Let $y \# \Psi, M, N, \widetilde{x}, \widetilde{T}, P$ be the tag used for encoding the choice. Then by definition of $\vdash_{\mathcal{E}}$ we have $(\Psi, \emptyset) \vdash_{\mathcal{E}} K \mathrel{\dot\to} M_y$. By rule In we can derive

$$(\Psi, \emptyset) \rhd \underline{M_y}(\lambda \widetilde{x})N.(\{y\} \mid [\![P]\!]) \xrightarrow[M_y]{\underline{K}\, N[\widetilde{x} := \widetilde{T}]} \{y\} \mid [\![P]\!][\widetilde{x} := \widetilde{T}]$$

By rule Par (letting $R = \alpha_y.(\{y\} \mid [\![Q]\!])$)

$$(\Psi, \emptyset) \rhd \underline{M_y}(\lambda \widetilde{x})N.(\{y\} \mid [\![P]\!]) \mid R \xrightarrow[M_y]{\underline{K}\, N[\widetilde{x} := \widetilde{T}]} \{y\} \mid [\![P]\!][\widetilde{x} := \widetilde{T}] \mid R$$

and by Res

$$(\Psi, \emptyset) \rhd [\![\underline{M}(\lambda \widetilde{x})N.P + \alpha.Q]\!] \xrightarrow[(\nu y; \epsilon)M_y]{\underline{K}\, N[\widetilde{x} := \widetilde{T}]} (\nu y)(\{y\} \mid [\![P]\!][\widetilde{x} := \widetilde{T}] \mid R)$$

Since $y \# \widetilde{x}, \widetilde{T}, P$ it follows that $y \# [\![P']\!][\widetilde{x} := \widetilde{T}]$. By scope extension, associativity, commutativity and Lemma 21 we have

$$(\nu y)(\{y\} \mid [\![P]\!][\widetilde{x} := \widetilde{T}] \mid R) \mathrel{\dot\sim}_{(\Psi,\emptyset)} [\![P]\!][\widetilde{x} := \widetilde{T}]$$

We may then conclude by using Lemma 20 to push the substitution inside the $[\![\_]\!]$ brackets.

2. By induction on the derivation of the transition from P. The proof is symmetric to the proof for input transitions.
3. By induction on the derivation of the transition from P. The interesting case is Com, where we know that $\Psi \otimes \Psi_Q \rhd P \xrightarrow[(\nu\widetilde{b}_P;\widetilde{y})K]{\overline{M}(\nu\widetilde{x})N} P'$ and $\Psi \otimes \Psi_P \rhd Q \xrightarrow[(\nu\widetilde{b}_Q;\widetilde{z})M]{\underline{K}\,N} Q'$ and $\mathcal{F}(P) = (\nu\widetilde{b}_P)\Psi_P$ and $\mathcal{F}(Q) = (\nu\widetilde{b}_Q)\Psi_Q$ and $\widetilde{b}_P\#\Psi, P, Q, M, \widetilde{z}, \widetilde{y}, \widetilde{b}_Q, \Psi_Q$ and $\widetilde{b}_Q\#\Psi, P, Q, b, \widetilde{z}, \widetilde{b}_Q, \Psi_P$ and $\widetilde{x}\#\Psi, P, Q$ and $\widetilde{z}, \widetilde{y}$ are similarly fresh.

   By Lemma 8.2 we obtain P'' such that $(\Psi, \emptyset) \otimes_{\mathcal{E}} (\Psi_Q, \emptyset) \rhd [\![P]\!] \xrightarrow[\pi]{\overline{M}(\nu\widetilde{x})N} P''$ and $P'' \mathrel{\dot{\sim}}_{(\Psi,\emptyset)\otimes_{\mathcal{E}}(\Psi_Q,\emptyset)} [\![P']\!]$. By Lemma 1 and Lemma 22 there are sufficiently fresh $\widetilde{a}, y$ such that $\pi = (\widetilde{b}_P, y; \widetilde{a})K'$ and $(\Psi \otimes \Psi_P \otimes \Psi_Q, \emptyset) \vdash K' \dot{\to} M$. By Lemma `Semantics.comm1_aux` we have

$$\Psi \otimes \Psi_P \rhd Q \xrightarrow[(\nu\widetilde{b}_Q;\widetilde{z})M]{\underline{K'_\perp}\,N} Q'$$

   By Lemma 8.1 we obtain Q'' such that $(\Psi, \emptyset) \otimes_{\mathcal{E}} (\Psi_P, \emptyset) \rhd [\![Q]\!] \xrightarrow[\pi']{\underline{K'_\perp}\,N} Q''$ and $Q'' \mathrel{\dot{\sim}}_{(\Psi,\emptyset)\otimes_{\mathcal{E}}(\Psi_P,\emptyset)} [\![Q']\!]$. By Lemma 1 and Lemma 22 there are sufficiently fresh $\widetilde{b}, z$ such that $\pi' = (\widetilde{b}_Q, z; \widetilde{a})M'$ and $(\Psi \otimes \Psi_P \otimes \Psi_Q, \emptyset) \vdash K' \dot{\to} M'$. By Lemmas `Semantics.comm1_aux` and `Semantics.comm2_aux` and the absence of disabled tags we may conclude

$$(\Psi, \emptyset) \otimes_{\mathcal{E}} (\Psi_P, \emptyset) \rhd [\![Q]\!] \xrightarrow[(\widetilde{b}_Q,z;\widetilde{a})M']{\underline{K'}\,N} Q''$$

   and $(\Psi, \emptyset) \otimes_{\mathcal{E}} (\Psi_Q, \emptyset) \rhd [\![P]\!] \xrightarrow[(\widetilde{b}_P,y;\widetilde{a})K']{\overline{M'}(\nu\widetilde{x})N} P''$ from which we may use the Com rule to infer

$$(\Psi, \emptyset) \rhd [\![P]\!] \mid [\![Q]\!] \xrightarrow[\perp]{\tau} (\nu\widetilde{x})(P'' \mid Q'')$$

   We may then derive $P'' \mid Q'' \mathrel{\dot{\sim}}_{\Psi} [\![P']\!] \mid [\![Q']\!]$ using Theorem 1, Lemma `Semantics.extend_frame` and extension of arbitrary assertion, and finally

$$(\nu\widetilde{x})(P'' \mid Q'') \mathrel{\dot{\sim}}_{\Psi} (\nu\widetilde{x})([\![P']\!] \mid [\![Q']\!])$$

   by Theorem 1.
4. By structural induction on P followed by inversion on the derivation of the transition from $[\![P]\!]$. We show only the cases of input and choice; the rest are routine because the encoding is homomorphic.

   $P = \underline{K}(\lambda\widetilde{x})L.Q$. Then the transition from $[\![P]\!] = \underline{K}(\lambda\widetilde{x})L.[\![Q]\!]$ must have been derived via the input rule, so $(\Psi, \mathbf{N}) \vdash_{\mathcal{E}} M \dot{\to} K$ and $N = L[\widetilde{x} := \widetilde{T}]$. Since K is not tagged we have $\Psi \vdash M_\perp \dot{\to} K$, and we may use the In rule to derive

$$\Psi \rhd P \xrightarrow[K]{\underline{M}\,N} Q[\widetilde{x} := \widetilde{T}]$$

whose encoding is bisimilar to the derivative of $[\![P]\!]$ because it's syntactically equal.

$P = \alpha.Q + \beta.R$. Then the transition from

$$[\![P]\!] = (\nu x)(\alpha_x.(\{x\} \mid Q) \mid \beta_x.(\{x\} \mid R))$$

with $x\#\alpha, Q, \beta, R, M, N, \Psi, \mathbf{N}$, must have been derived via Res, Par and In. Assume α_x is the prefix used to derive the transition (the case for β_x is symmetric). Then there exists $K, \widetilde{x}, \widetilde{T}, L$ such that $\alpha = \underline{K_X}(\lambda \widetilde{x} L)$ and $N = L[\widetilde{x} := \widetilde{T}]$ and $(\Psi, \mathbf{N}) \vdash_{\mathcal{E}} M \leftrightarrows K_x$ and

$$(\Psi, \mathbf{N}) \rhd \alpha.Q \xrightarrow[K_x]{\underline{M}\, N} \{x\} \mid Q[\widetilde{x} := \widetilde{T}]$$

and, by rules Res and Par,

$$(\Psi, \mathbf{N}) \rhd [\![P]\!] \xrightarrow[K_x]{\underline{M}\, N} (\nu x)(\{x\} \mid [\![Q]\!][\widetilde{x} := \widetilde{T}] \mid \beta_x.(\{x\} \mid [\![R]\!]))$$

By scope extension and Lemma 21,

$$(\nu x)(\{x\} \mid Q[\widetilde{x} := \widetilde{T}] \mid \beta_x.(\{x\} \mid R)) \mathrel{\dot{\sim}}_{(\Psi, \mathbf{N})} [\![Q[\widetilde{x} := \widetilde{T}]]\!]$$

The matching transition from P is, by Case and In and $\Psi \vdash M \leftrightarrows K$,

$$\Psi \rhd \alpha.Q + \beta.R \xrightarrow[K]{\underline{M}\, N} Q[\widetilde{x} := \widetilde{T}]$$

which suffices.

5. By structural induction on P followed by inversion on the derivation of the transition from $[\![P]\!]$. Similar to the case for input.
6. By structural induction on P followed by inversion on the derivation of the transition from $[\![P]\!]$. The interesting case is $P = \alpha.Q + \beta.R$, where we need to show the absence of a communication between α_y and β_y in $[\![P]\!]$. This is immediate from Lemma 1 and the definition of $\vdash_{\mathcal{E}}$, which requires distinct tags.

Lemma 23. – *If* $(\Psi, \mathbf{N}) \rhd P \xrightarrow{\alpha_x} \pi P'$ *and* $y \notin \mathbf{N}$ *then* $(\Psi, \mathbf{N}) \rhd P \xrightarrow{\alpha_y} \pi P'$ *and* $x \notin \mathbf{N}$.
– *If* $(\Psi, \mathbf{N}) \rhd P \xrightarrow{\alpha_\perp} \pi P'$ *and* $y \notin \mathbf{N}$ *then* $(\Psi, \mathbf{N}) \rhd P \xrightarrow{\alpha_y} \pi P'$
– *If* $(\Psi, \mathbf{N}) \rhd P \xrightarrow{\alpha_x} \pi P'$ *then* $(\Psi, \mathbf{N}) \rhd P \xrightarrow{\alpha_\perp} \pi P'$ *and* $x \notin \mathbf{N}$.

Proof. Follows from Lemma 1, `Semantics.transfer_frame` and the definition of $\vdash_{\mathcal{E}}$.

Definition 19. *A process P is* untagged *if it contains no free tags in subjects, assertions or conditions, and if no name (free or bound) occurs both in the tags of P and in the non-tags of P.*

Lemma 24. $[\![P]\!]$ *is untagged.*

Proof. By structural induction on P.

Lemma 25. *If P is untagged and $(\Psi, \mathbf{N}) \rhd P \xrightarrow[\pi]{\alpha_\perp} Q$, then $(\Psi, \mathbf{N}') \rhd P \xrightarrow[\pi]{\alpha_\perp} Q$ and Q is untagged.*

Proof. By induction on the derivation of the transition from P. In the Open rule we use the fact that since tags cannot occur in output objects, the name being exported does not occur in tags.

Lemma 26. *If P, Q are untagged then $(\Psi, \mathbf{N}) \rhd P \mathrel{\dot{\sim}} Q$ iff $(\Psi, \mathbf{N}') \rhd P \mathrel{\dot{\sim}} Q$*

Proof. By showing that $\mathcal{R} = \{((\Psi, \mathbf{N}), P, Q).P \mathrel{\dot{\sim}}_{(\Psi,\mathbf{N}')} Q \wedge P, Q \text{ untagged}\}$ is a bisimulation relation. Extension of arbitrary assertion and symmetry are trivial. Static equivalence follows by Theorem 18.

For simulation, we have that $(\Psi, \mathbf{N}) \rhd P \xrightarrow[\pi]{\alpha} P'$ and $P \mathrel{\dot{\sim}}_{(\Psi,\mathbf{N}')} Q$. By Lemma 23, $(\Psi, \mathbf{N}) \rhd \alpha_\perp \xrightarrow[P']{\pi}$. By Lemma 25, $(\Psi, \mathbf{N}') \rhd P \xrightarrow[\pi]{\alpha_\perp} P'$ and P' is untagged. By simulation there is Q' such that $(\Psi, \mathbf{N}') \rhd Q \xrightarrow[\pi']{\alpha_\perp} Q'$ and $P' \mathrel{\dot{\sim}}_{(\Psi,\mathbf{N}')} Q'$. By Lemma 25, $(\Psi, \mathbf{N}) \rhd Q \xrightarrow[\pi']{\alpha_\perp} Q'$ and Q' is untagged. Finally by Lemma 23, $(\Psi, \mathbf{N}) \rhd Q \xrightarrow[\pi']{\alpha} Q'$.

Lemma 27. *If $P \mathrel{\dot{\sim}}_\Psi Q$ then $[\![P]\!] \mathrel{\dot{\sim}}_{(\Psi,\emptyset)} [\![Q]\!]$.*

Proof. We show that the relation $\mathcal{R} = \{(\Psi, P, Q) : [\![P]\!] \mathrel{\dot{\sim}}_{(\Psi,\emptyset)} [\![Q]\!]\}$ is a bisimulation relation.

- Static equivalence: follows from Lemma 22 and Lemma 18.
- Symmetry follows by symmetry of $\mathrel{\dot{\sim}}$.
- Extension of arbitrary assertion follows by the corresponding property of $\mathrel{\dot{\sim}}$.
- Simulation: We have $\Psi \rhd P \xrightarrow{\alpha} P'$ and $[\![P]\!] \mathrel{\dot{\sim}}_{(\Psi,\emptyset)} [\![Q]\!]$. By Lemma 8 there exists P'' such that $(\Psi, \emptyset) \rhd [\![P]\!] \xrightarrow{\alpha} P''$ and $P'' \mathrel{\dot{\sim}}_{(\Psi,\emptyset)} [\![P']\!]$. By simulation there is Q'' such that $(\Psi, \emptyset) \rhd [\![Q]\!] \xrightarrow{\alpha} Q''$ $P'' \mathrel{\dot{\sim}}_{(\Psi,\emptyset)} Q''$. By Lemma 8 and since $\alpha = \alpha_\perp$ there is Q' such that $Q' \mathrel{\dot{\sim}}_{(\Psi,\emptyset)} [\![Q'']\!]$ and $\Psi \rhd Q \xrightarrow{\alpha} Q'$. The desired result follows by transitivity and symmetry of $\mathrel{\dot{\sim}}$.

Lemma 28. *If $[\![P]\!] \mathrel{\dot{\sim}}_{(\Psi,\mathbf{N})} [\![Q]\!]$ then $P \mathrel{\dot{\sim}}_\Psi Q$.*

Proof. We show that the relation $\mathcal{R} = \{((\Psi, \mathbf{N}), [\![P]\!], [\![Q]\!]) : P \mathrel{\dot{\sim}}_{(\Psi} Q\}$ is a bisimulation up to $\mathrel{\dot{\sim}}$.
Static equivalence: follows from Lemma 22 and Lemma 18.
Symmetry follows by symmetry of $\mathrel{\dot{\sim}}$.
Extension of arbitrary assertion follows by the corresponding property of $\mathrel{\dot{\sim}}$.
Simulation. $(\Psi, \mathbf{N}) \rhd [\![P]\!] \xrightarrow{\alpha} P''$ and $P \mathrel{\dot{\sim}}_\Psi Q$. By Lemma 8 there exists P' such that $\Psi \rhd P \xrightarrow{\alpha_\perp} P'$ and $[\![P']\!] \mathrel{\dot{\sim}}_{(\Psi,\mathbf{N})} P''$. By simulation there is Q'

such that $P' \mathrel{\dot\sim_\Psi} Q'$ and $\Psi \rhd Q \xrightarrow{\alpha_\perp} Q'$. By Lemma 8 there is Q'' such that $[\![Q']\!] \mathrel{\dot\sim_{(\Psi,\emptyset)}} Q''$ and $(\Psi,\emptyset) \rhd [\![Q]\!] \xrightarrow{\alpha_\perp} Q''$. By extension of arbitrary assertion, by Lemmas 23–25 and since $[\![Q]\!], [\![Q']\!], Q''$ are untagged, we conclude $[\![Q']\!] \mathrel{\dot\sim_{(\Psi,\mathbf{N})}} Q''$ and $(\Psi,\mathbf{N}) \rhd [\![Q]\!] \xrightarrow{\alpha_\perp} Q''$.

Reduction semantics

Lemma 29. *$\equiv$ is a bisimulation relation.*

Proof. Similar to the proofs of Theorems 1–2.

Lemma 30. *If $\widetilde{\Psi} \vdash M \doteq N$ and $\forall \varphi \in \mathrm{conds}(C).\ \widetilde{\Psi} \vdash \varphi$ then*

1. *There is P' such that $\widetilde{\Psi} \rhd C[\overline{M}\, K.P] \xrightarrow[M]{\overline{N} K} P'$. and $P' \equiv P \mid \mathrm{ppr}(C)$.*
2. *There is P' such that $\widetilde{\Psi} \rhd C[\underline{M}(\lambda \widetilde{x})L.P] \xrightarrow[M]{\underline{N}\, L[\widetilde{x}:=\widetilde{T}]} P'$ and $P' \equiv P[\widetilde{x} := \widetilde{T}] \mid \mathrm{ppr}(C)$.*

Proof. By a straightforward structural induction on P.

Lemma 31. *If $P \longrightarrow P'$ then there is P'' such that $\mathbf{1} \rhd P \xrightarrow{\tau} P''$ and $P'' \equiv P'$.*

Proof. By induction on the derivation of $P \longrightarrow P'$. There are three cases:

Struct We know that $P \equiv Q$ and $Q \longrightarrow Q'$ and $P \longrightarrow P'$. By the induction hypothesis, $\mathbf{1} \rhd P \xrightarrow{\tau} P''$ and $P'' \equiv P'$. Since $\equiv$ is a bisimulation relation there is Q'' such that $\mathbf{1} \rhd Q \xrightarrow{\tau} Q''$ and $P'' \equiv Q''$. $Q'' \equiv P'$ follows by symmetry and transitivity of $\equiv$.

Scope We have $P \longrightarrow P'$ and, by the induction hypothesis, $\mathbf{1} \rhd P \xrightarrow{\tau} P''$ and $P'' \equiv P'$. By rule Scope, $\mathbf{1} \rhd (\nu x)P \xrightarrow{\tau} (\nu x)P''$, and because $\equiv$ is a congruence, $(\nu x)P'' \equiv (\nu x)P'$.

Ctxt We have $\widetilde{\Psi} \vdash M \doteq N$ and $\forall \varphi \in \mathrm{conds}(C).\ \widetilde{\Psi} \vdash \varphi$ and By structural induction on C (using Lemma 30 to derive a communication in the base case) we can construct a derivation

$$\widetilde{\Psi} \rhd C[\overline{M}\, K.P,\ \underline{N}(\lambda \widetilde{x})L.Q] \xrightarrow{\tau} P'$$

such that $P' \equiv P \mid Q[\widetilde{x} := \widetilde{T}] \mid \mathrm{ppr}(C)$. By rule Par and closure of $\equiv$ under parallel composition,

$$\mathbf{1} \rhd \widetilde{(\!|\Psi|\!)} \mid C[\overline{M}\, K.P,\ \underline{N}(\lambda \widetilde{x})L.Q] \xrightarrow{\tau} \widetilde{(\!|\Psi|\!)} \mid P'$$

and $\widetilde{(\!|\Psi|\!)} \mid P' \equiv \widetilde{(\!|\Psi|\!)} \mid P \mid Q[\widetilde{x} := \widetilde{T}] \mid \mathrm{ppr}(C)$.

Lemma 32. *For every process P, there exists $\widetilde{x}, \widetilde{\Psi}, P_G$ such that*

$$P \equiv (\nu\widetilde{x})(\widetilde{(\!|\Psi|\!)} \mid P_G)$$

and P_G contains no ν-binders except underneath replication, input or output operators.

Proof. By a straightforward structural induction on P.

Definition 20. *The* normalisation *of a process P, written $P^{\downarrow}$, is defined homomorphically on all operators except $!P$, where it is defined as $(!P)^{\downarrow} = P^{\downarrow} \mid P^{\downarrow} \mid !P$.*

Lemma 33.

1. $P_G \equiv P_G^{\downarrow}$
2. *If $\Psi \rhd P_G \xrightarrow[\pi]{\alpha} P'$ then there is P'' such that $P' \equiv P''$ and there is a derivation of $\Psi \rhd P_G^{\downarrow} \xrightarrow[\pi]{\alpha} P''$ that does not use the* Rep *rule.*

Proof.

1. By structural induction on P_G. The only non-trivial case is when $P_G = !Q_G$. Unfolding the replication twice yields $!Q_G \equiv Q_G \mid Q_G \mid !Q_G$, and by the induction hypothesis $Q_G \equiv Q_G^{\downarrow}$. We can then derive

$$(!Q_G)^{\downarrow} = Q_G^{\downarrow} \mid Q_G^{\downarrow} \mid !Q_G \equiv Q_G \mid Q_G \mid !Q_G \equiv !Q_G$$

2. By structural induction on P_G. Again, the non-trivial case is when $P_G = !Q_G$. There are two cases: either the transition originates from a single copy of Q_G, or from a communication between two copies of Q_G. By the induction hypothesis we can derive the same transitions from $Q_G^{\downarrow}$, which allows the matching derivation from $(!Q_G)^{\downarrow}$ to use its two top-level copies of $Q_G^{\downarrow}$ instead of unfolding the replication.

Lemma 34. *If P_G contains no ν-binders except underneath replication, input and output prefixes, and if there is a derivation of $\Psi \rhd P_G \xrightarrow[\pi]{\alpha} P'$ that does not use the* Rep *rule, then*

1. *If $\alpha = \overline{M}N$ and $\pi = K$ then there is C, Q such that $P_G = C[\overline{K}\,N.Q]$ and $\forall\varphi \in \mathrm{conds}(C).\ \widetilde{\Psi} \vdash \varphi$ and $P' \equiv Q \mid \mathrm{ppr}(C)$.*
2. *If $\alpha = \underline{M}\,N$ and $\pi = K$ then there is $C, Q, L, \widetilde{x}, \widetilde{T}$ such that $N = L[\widetilde{x} := \widetilde{T}]$ and $P_G = C[\underline{K}(\lambda\widetilde{x})L.Q]$ and $\forall\varphi \in \mathrm{conds}(C).\ \widetilde{\Psi} \vdash \varphi$ and $P' \equiv Q[\widetilde{x} := \widetilde{T}] \mid \mathrm{ppr}(C)$.*
3. *If $\alpha = \tau$ then there is $C, Q, R, M, L, \widetilde{x}, \widetilde{T}$ such that $\forall\varphi \in \mathrm{conds}(C).\ \widetilde{\Psi} \vdash \varphi$ and $P' \equiv Q \mid R[\widetilde{x} := \widetilde{T}] \mid \mathrm{ppr}(C)$ and $\widetilde{\Psi} \vdash M \leftrightarrows K$ and either $P_G = C[\overline{M}\,L[\widetilde{x} := \widetilde{T}].Q, \underline{K}(\lambda\widetilde{x})L.R]$ or $P_G = C[\underline{K}(\lambda\widetilde{x})L.R, \overline{M}\,L[\widetilde{x} := \widetilde{T}].Q]$*
4. *Neither π nor α have any bound names.*

Proof.

1. Similar to, but easier than, the case for input below.
2. By induction on the derivation of $\Psi \rhd P_G \xrightarrow[\pi]{\alpha} P'$. Note that rules Rep and Scope cannot be used in this derivation, and because P_G is guarded there are no additional frames to consider in the Par rules. The cases to consider are:

   In Immediate by choosing $C = [\,]$.

   Case We know that $\Psi \rhd P_i \xrightarrow[\pi]{M\,N} P'$ and $\Psi \vdash \varphi_i$. By the induction hypothesis there is $C', Q, L, \widetilde{X}, \widetilde{T}$ such that $N = L[\widetilde{x} := \widetilde{T}]$ and $P_i = C'[\underline{K}(\lambda\widetilde{x})L.Q]$ and $P'_i \equiv Q[\widetilde{x} := \widetilde{T}] \mid \mathrm{ppr}(C')$ $\forall \varphi \in \mathrm{conds}(C').\ \widetilde{\Psi} \vdash \varphi$.
   We can close the proof by choosing $C = \mathbf{case}\ \widetilde{\varphi} : \widetilde{P_G}\ [\!]\ \varphi_i : C'\ [\!]\ \widetilde{\varphi'} : \widetilde{Q_G}$: we then have $\mathrm{ppr}(C) = \mathrm{ppr}(C')$ and $\mathrm{conds}(C) = \mathrm{conds}(C') \cup \varphi_i$, all of whom are entailed by Ψ.

   Par-L We know that $\Psi \rhd P \xrightarrow[\pi]{\underline{M}\,N} P'$. By the induction hypothesis there is $C', Q, L, \widetilde{X}, \widetilde{T}$ such that $N = L[\widetilde{x} := \widetilde{T}]$ and $P = C'[\underline{K}(\lambda\widetilde{x})L.Q]$ and $P' \equiv Q[\widetilde{x} := \widetilde{T}] \mid \mathrm{ppr}(C')$.
   We can close the proof by choosing $C = C' \mid R$. It follows that $\mathrm{ppr}(C) = \mathrm{ppr}(C') \mid R$ and thus $P' \mid R \equiv Q[\widetilde{x} := \widetilde{T}] \mid \mathrm{ppr}(C)$.

   Par-R Symmetric to Par-L.
3. By induction on the derivation of $\Psi \rhd P_G \xrightarrow[\pi]{\alpha} P'$. The proof is similar to the input case. The only interesting difference is the Com case, where we use Lemma 34.1–2 to obtain contexts $P = C'[\overline{M}\,N.Q]$ and $R = C''[\underline{K}(\lambda\widetilde{x})L.S]$ for the two communication partners. To close the proof we choose $C = C' \mid C''$.
4. Any binders in π or α must have been introduced by the rules Par, Scope or Open. Since P_G has no binders except underneath prefixes, the derivation of the transition does not use rules Scope nor Open, and the Par rule is only applied to processes with empty frame binding sequences.

Lemma 35. $C[P, Q] \equiv C[Q, P]$

Proof. By induction on C.

Lemma 36. *If* $\mathbf{1} \rhd P \xrightarrow{\tau} P'$ *then* $P \longrightarrow P'$

Proof. By Lemma 32 there is $\widetilde{y}, \widetilde{\Psi}, P_G$ such that $P \equiv (\nu\widetilde{y})(\widetilde{(\!|\Psi|\!)} \mid P_G)$ and P_G contains no ν-binders except underneath replication, input or output operators. By Lemma 29 there is P'_G such that $\mathbf{1} \rhd (\nu\widetilde{y})(\widetilde{(\!|\Psi|\!)} \mid P_G) \xrightarrow{\tau} P'_G$ and $P' \equiv P'_G$. Case analysis on the derivation of this transition yields that there must be P''_G such that $P'_G = (\nu\widetilde{y})(\widetilde{(\!|\Psi|\!)} \mid P''_G)$ and $\widetilde{\Psi} \rhd P_G \xrightarrow{\tau} P'_G$. By Lemma 33 we have $P_G \equiv P_G^{\downarrow}$ and obtain P'''_G such that $P''_G \equiv P'''_G$ and $\widetilde{\Psi} \rhd P_G^{\downarrow} \xrightarrow{\tau} P'''_G$ through a derivation that does not use the Rep rule. By Lemmas 34.3 and 35 there is $C, \widetilde{Q}, R, M, L, \widetilde{x}, \widetilde{T}$ such that $\forall \varphi \in \mathrm{conds}(C).\ \widetilde{\Psi} \vdash \varphi$ and $P'''_G \equiv Q \mid R[\widetilde{x} := \widetilde{T}] \mid \mathrm{ppr}(C)$ and

$\widetilde{\Psi} \vdash M \dot{\to} K$ and $P_G^{\downarrow} = C[\overline{M}\, L[\widetilde{x} := \widetilde{T}].Q, \underline{K}(\lambda\widetilde{x})L.R]$. This allows us to infer the desired reduction as follows (where Res is applied $|\widetilde{y}|$ times):

$$\textsc{Struct}\ \frac{\textsc{Res}\ \dfrac{\textsc{Struct}\ \dfrac{\textsc{Ctxt}\ \dfrac{\widetilde{\Psi} \vdash M \dot{\to} K \qquad \forall \varphi \in \mathrm{conds}(C).\ \widetilde{\Psi} \vdash \varphi}{\widetilde{(\!|\Psi|\!)} \mid P_G^{\downarrow} \ \longrightarrow\ \widetilde{(\!|\Psi|\!)} \mid Q \mid R[\widetilde{x} := \widetilde{T}] \mid \mathrm{ppr}(C)}}{\widetilde{(\!|\Psi|\!)} \mid P_G \ \longrightarrow\ \widetilde{(\!|\Psi|\!)} \mid P_G''}}{(\nu\widetilde{y})(\widetilde{(\!|\Psi|\!)} \mid P_G) \ \longrightarrow\ (\nu\widetilde{y})(\widetilde{(\!|\Psi|\!)} \mid P_G'')}}{P \ \longrightarrow\ P'}$$

Barbed bisimulation

Lemma 37. $\equiv \subseteq \dot{\sim}_{\mathbf{1}}$

Proof. Immediate from Theorems 1 and 2.

Lemma 38. $\equiv \subseteq \underset{\text{barb}}{\dot{\sim}}$

Proof. By coinduction, using $\equiv$ as candidate relation. Barb similarity is immediate from Lemma 37. Reduction simulation follows from rule Struct. Closure under static contexts holds by definition of $\equiv$

Lemma 39 (Soundness). *$P \dot{\sim}_{\mathbf{1}} Q$ implies $P \underset{\text{barb}}{\dot{\sim}} Q$.*

Proof. By coinduction, using candidate relation $\dot{\sim}_{\mathbf{1}}$. We have three cases:

- *(barb similarity)* We have that $P \dot{\sim}_{\mathbf{1}} Q$, $P \downarrow_{\overline{M}(\nu\widetilde{a})N}$, and $\widetilde{a} \# Q$. By definition of $\downarrow$, there is P' such that $\mathbf{1} \rhd P \xrightarrow{\overline{M}(\nu\widetilde{a})N} P'$. By simulation, $\exists Q'.\mathbf{1} \rhd Q \xrightarrow{\overline{M}(\nu\widetilde{a})N} Q'$, which by definition of $\downarrow$ is $Q \downarrow_{\overline{M}(\nu\widetilde{a})N}$.
- *(reduction simulation)* We have $P \dot{\sim}_{\mathbf{1}} Q$ and $P \longrightarrow P'$. By Theorem 4 there is P'' such that $P'' \equiv P'$ and $\mathbf{1} \rhd P \xrightarrow{\tau} P''$. By simulation there is Q' such that $\mathbf{1} \rhd Q \xrightarrow{\tau} Q'$ and $P'' \dot{\sim}_{\mathbf{1}} Q'$. Theorem 4 gives $Q \xrightarrow{\tau} Q'$, and $P' \dot{\sim}_{\mathbf{1}} Q'$ follows by transitivity of $\dot{\sim}$ and Lemma 37.
- *(closure under static contexts)* We have $P \dot{\sim}_{\mathbf{1}} Q$. By Theorem 1.1 we get and $P \mid R \dot{\sim}_{\mathbf{1}} Q \mid R$. Then $(\nu\widetilde{a})(P \mid R) \dot{\sim}_{\mathbf{1}} (\nu\widetilde{a})(Q \mid R)$ follows by induction on the length of $\widetilde{a}$, using Theorem 1.2 and equivariance of $\mathbf{1}$.

Lemma 40. *In all observational psi-calculi, $P \mid (\!|\Psi|\!) \underset{\text{barb}}{\dot{\sim}} Q \mid (\!|\Psi|\!)$ implies $\Psi \otimes \mathcal{F}(P) \simeq \Psi \otimes \mathcal{F}(Q)$*

Proof. By contradiction. Assume $\Psi \otimes \mathcal{F}(P) \not\simeq \Psi \otimes \mathcal{F}(Q)$. It follows, without loss of generality, that there is φ such that $\Psi \otimes \mathcal{F}(P) \vdash \varphi$ but not $\Psi \otimes \mathcal{F}(Q) \vdash \varphi$. By observationality, we obtain M_P, K_P such that $\mathcal{F}(P \mid (\!|\Psi|\!)) \simeq \Psi \otimes \mathcal{F}(P) \vdash M_P \dot{\to} K_P$ and not $P \mid (\!|\Psi|\!) \downarrow_{\overline{K_p} K_p}$.

Let $R = \mathbf{case}\ \varphi : \overline{M_P}\, K_P.0$ By closure under static contexts, $P \mid R \underset{\text{barb}}{\dot{\sim}} Q \mid R$. Using the ParR, Case and Out rules we can derive $\mathbf{1} \rhd P \mid R \xrightarrow[M_p]{\overline{K_P} K_P} P \mid (\!|\Psi|\!) \mid 0)$; hence $P \mid R \downarrow_{\overline{K_P} K_P}$. By barb similarity we also have $Q \mid R \downarrow_{\overline{K_P} K_P}$ and by definition of $\downarrow$ there is π, Q' such that $\mathbf{1} \rhd Q \mid R \xrightarrow[\pi]{\overline{K_P} K_P} Q'$.

The proof proceeds by case analysis on the derivation of this transition, deriving a contradiction in each case. We have two subcases depending on which parallel component the output originates from:

- (ParL) We have $\Psi \rhd Q \xrightarrow[\pi]{\overline{K_P} K_P} Q''$ for some Q''. By rule ParL and the definition of $\downarrow$ this implies $Q \mid (\!|\Psi|\!) \downarrow_{\overline{K_P} K_P}$. But by barb similarity and symmetry of $\underset{\text{barb}}{\dot{\sim}}$ we have $P \mid (\!|\Psi|\!) \downarrow_{\overline{K_P} K_P}$ which contradicts observationality.
- (ParR) We have $\Psi \otimes \Psi_Q \rhd R \xrightarrow[\pi]{\overline{K_P} K_P} Q''$ for some $Q'', \Psi_Q, \widetilde{b}_Q$ such that $\widetilde{b}_Q \# \varphi, M_P, K_P, \Psi$. This transition can only be derived by the Case rule, so $\Psi \otimes \Psi_Q \vdash \varphi$. But this means that $(\nu \widetilde{b}_Q)\Psi \otimes \Psi_Q \simeq \Psi \otimes \mathcal{F}(Q) \vdash \varphi$, which is a contradiction.

Lemma 41. $(\!|\Psi|\!) \mid (\!|\Psi'|\!) \underset{\text{barb}}{\dot{\sim}} (\!|\Psi \otimes \Psi'|\!)$

Proof. By soundness it suffices to show that the symmetric closure of

$$\mathcal{R} = \bigcup_{\Psi''} \{(\Psi'', (\!|\Psi|\!) \mid (\!|\Psi'|\!), (\!|\Psi \otimes \Psi'|\!))$$

is a bisimulation relation. Symmetry and extension of arbitrary assertion hold by construction, simulation because neither agent has any transitions, and static equivalence because $\mathcal{F}((\!|\Psi|\!) \mid (\!|\Psi'|\!)) = \mathcal{F}((\!|\Psi \otimes \Psi'|\!)) = \Psi \otimes \Psi'$.

Lemma 42. *In all observational psi-calculi: suppose* $\widetilde{x} \subseteq \widetilde{N}$ *and* $\widetilde{x} \# M_P$ *and* $(\nu \widetilde{x})(P \mid \overline{M_P} N \underset{\text{barb}}{\dot{\sim}} (\nu \widetilde{x})(Q \mid \overline{M_P} N)$ *and not* $P \Downarrow_{\overline{M_p}}$. *Then* $P \underset{\text{barb}}{\dot{\sim}} Q$.

Proof. (Sketch) The proof is a direct adaptation of [14, Lemma 6.61] to the case of strong barbed bisimilarity.

Lemma 43 (Completeness). *In all observational psi-calculi,* $P \underset{\text{barb}}{\dot{\sim}} Q$ *implies* $P \dot{\sim}_{\mathbf{1}} Q$.

Proof. We show that $\mathcal{R} = \{(\Psi, P, Q) : P \mid (\!|\Psi|\!) \underset{\text{barb}}{\dot{\sim}} Q \mid (\!|\Psi|\!)\}$ is a bisimulation relation.

- *(static equivalence)* By Lemma 40.
- *(symmetry)* By symmetry of $\underset{\text{barb}}{\dot{\sim}}$.

– *(simulation)* We have $\Psi \rhd P \xrightarrow[\pi]{\alpha} P'$, $P \mid (\!|\Psi|\!) \mathrel{\dot{\sim}}_{\text{barb}} Q \mid (\!|\Psi|\!)$ and $\text{bn}(\alpha)\#Q$. By observationality we obtain M_P, K_P and $M_{P'}, K_{P'}$ such that $\Psi \otimes \mathcal{F}(P) \vdash M_P \dot{\to} K_P$ and $\Psi \otimes \mathcal{F}(P') \vdash M_{P'} \dot{\to} K_{P'}$ but neither $P \mid (\!|\Psi|\!) \downarrow_{\overline{K_P} K_P}$ nor $P' \mid (\!|\Psi|\!) \downarrow_{\overline{K_{P'}} K_{P'}}$.
The proof proceeds by case analysis on what kind of label α is.

- $\alpha = \tau$. By rule ParL and Lemma 4, $P \mid \Psi \longrightarrow P' \mid \Psi$. By reduction simulation, $Q \mid (\!|\Psi|\!) \longrightarrow Q'$ and $P' \mid (\!|\Psi|\!) \mathrel{\dot{\sim}}_{\text{barb}} Q'$. By Lemma 4 there is Q'' such that $Q'' \equiv Q'$ and $\mathbf{1} \rhd Q \mid (\!|\Psi|\!) \xrightarrow{\tau} Q''$. This transition must have been derived from rule ParL; hence $Q'' = Q''' \mid (\!|\Psi|\!)$ for some $\Psi \rhd Q \xrightarrow{\tau} Q'''$. By Lemma 38 and transitivity of $\mathrel{\dot{\sim}}_{\text{barb}}$ we have $P' \mid (\!|\Psi|\!) \mathrel{\dot{\sim}}_{\text{barb}} Q''' \mid (\!|\Psi|\!)$. Finally, $Q \mid (\!|\Psi|\!) \longrightarrow Q''' \mid (\!|\Psi|\!)$ follows by rule Struct.
- $\alpha = \underline{M}\, N$. By Lemma 1 there is $\widetilde{x}, \widetilde{b}_P, \Psi_P, K$ such that $\mathcal{F}(P) = (\nu \widetilde{b}_P)\Psi_P$ and $\pi = (\nu \widetilde{b}_P; \widetilde{x})K$ and $\widetilde{b}_P \# \Psi, P, M, N, P', Q, M_P, K_P, M_{P'}, K_{P'}, \widetilde{x}$ and $\widetilde{x} \# \Psi, P, N, Q, P', M_P, K_P, M_{P'}, K_{P'}$ and $\Psi \otimes \Psi_P \vdash M \dot{\to} K$. Let

$$R = \mathbf{case}\ M_P \dot{\to} K_P : \overline{M}\, N.0 \,[]\, M_{P'} \dot{\to} K_{P'} : \overline{M_{P'}}\, K_{P'}.\overline{M_{P'}}\, K_{P'}.0$$

  By rules Out and Case we can derive $\Psi \otimes \Psi_P \rhd R \xrightarrow[M]{\overline{K} N} 0$ and from Com, Lemma 4 and Struct, $P \mid (\!|\Psi|\!) \mid R \longrightarrow P' \mid (\!|\Psi|\!)$ follows.
  From closure under static contexts and reduction simulation, we obtain Q' such that $Q \mid (\!|\Psi|\!) \mid R \longrightarrow Q'$ and $P' \mid (\!|\Psi|\!) \mathrel{\dot{\sim}}_{\text{barb}} Q'$. From Lemma 4 we obtain $Q'' \equiv Q'$ such that $\mathbf{1} \rhd Q \mid (\!|\Psi|\!) \mid R \xrightarrow[\perp]{\tau} Q''$. The proof proceeds by case analysis on how this transition was derived; there are three cases to consider.
  - From an internal transition within Q; that is, $Q'' = Q''' \mid (\!|\Psi|\!) \mid R$ and $\Psi \otimes \Psi_R \rhd Q \xrightarrow[\perp']{\tau} Q'''$ for some Q'''. This, however, contradicts barb similarity: with ParR, Case, Out and Lemma 40 we can derive $Q''' \mid (\!|\Psi|\!) \mid R \downarrow_{\overline{K_{P'}} K_{P'}}$ but $P' \mid (\!|\Psi|\!)$ does not expose this barb.
  - From a communication between the prefix $\overline{M_{P'}} K_{P'}$ in R and some prefix in Q; that is, $Q'' = Q''' \mid (\!|\Psi|\!) \mid \overline{M_{P'}}\, K_{P'}$ and $\Psi \rhd Q \xrightarrow[(\nu \widetilde{x}; \widetilde{y})L]{\underline{M_{P'}}\, K_{P'}} Q'''$ for some $Q''', \widetilde{x}, , \widetilde{y}, L$. This contradicts barb similarity: with ParR, Out and Lemma 40 we can derive

$$Q''' \mid (\!|\Psi|\!) \mid \overline{M_{P'}}\, K_{P'} \downarrow_{\overline{K_{P'}} K_{P'}}$$

    but $P' \mid (\!|\Psi|\!)$ does not expose this barb.
  - From a communication between the prefix $\overline{M} N$ in R and some prefix in Q; that is, $Q'' = Q''' \mid (\!|\Psi|\!) \mid 0$ and $\Psi \rhd Q \xrightarrow[(\nu \widetilde{x}; \widetilde{y})L]{\underline{M}\, N} Q'''$ for some

$Q''', \widetilde{x}, , \widetilde{y}, L$. By Lemma 40 and transitivity of $\dot{\sim}_{\text{barb}}$ we can conclude $P' \dot{\sim}_{\text{barb}} Q''' \mid (\!|\Psi|\!)$.

- $\alpha = \overline{M}(\nu\widetilde{x})N$. By Lemma 1 there is $\widetilde{y}, \widetilde{b}_P, \Psi_P, K$ such that $\mathcal{F}(P) = (\nu\widetilde{b}_P)\Psi_P$ and $\pi = (\nu\widetilde{b}_P;\widetilde{y})K$ and $\widetilde{b}_P\#\Psi, P, M, N, P', Q, M_P, K_P, M_{P'}, K_{P'}, \widetilde{x}, \widetilde{y}$ and $\widetilde{y}\#\Psi, P, N, Q, P', M_P, K_P, M_{P'}, K_{P'}, \widetilde{x}$ and $\Psi \otimes \Psi_P \vdash K \dot{\to} M$.
  Let $\widetilde{z}, L, p$ be such that $p = (\widetilde{z}\ \widetilde{x})$ and $N = p \cdot L$ and

$$\widetilde{z}\#\Psi, P, M, K, N, P', Q, M_P, K_P, M_{P'}, K_{P'}, \widetilde{x}, \widetilde{y}.$$

  We will use the observing context

$$R = \underline{M}(\lambda\widetilde{z})L.\overline{p \cdot M_{P'}}\, L$$

  By observationality we know that $L[\widetilde{x} := \widetilde{y}] = N$ and $(p \cdot M'_P)[\widetilde{x} := \widetilde{y}] = M'_P$. Thus by rule In we can derive $\Psi \otimes \Psi_P \rhd R \xrightarrow[M]{\underline{K}\ N} \overline{M_{P'}}\, N$, and from Com, Lemma 4 and Struct,

$$P \mid (\!|\Psi|\!) \mid R \longrightarrow (\nu\widetilde{x})(P' \mid \overline{M_{P'}}\, N \mid (\!|\Psi|\!))$$

  From closure under static contexts and reduction simulation, we obtain Q' such that $Q \mid (\!|\Psi|\!) \mid R \longrightarrow Q'$ and $(\nu\widetilde{x})(P' \mid \overline{M_{P'}}\, N) \dot{\sim}_{\text{barb}} Q'$. By barb similarity and a similar case analysis to the case for input, we know that since $P' \Downarrow_{\overline{K_{P'}}\, N}$ it must be that $Q' \Downarrow_{\overline{K_{P'}}\, N}$; however, this is only possible if the reduction to Q' was derived from a communication between Q and R exposing said barb. That is, Q sends to M an object $(\nu\widetilde{x})N'$ such that $L[\widetilde{z} := \widetilde{T}] = N'$ and $L[\widetilde{z} := \widetilde{T}] = N$ and $(p \cdot M_{P'})[\widetilde{z} := \widetilde{T}] = N$, yielding $N' = N$. Thus there is Q'' such that

$$Q' \equiv (\nu\widetilde{x})(Q'' \mid \overline{M_{P'}}\, N \mid (\!|\Psi|\!))$$

  and

$$\Psi \rhd Q \xrightarrow{\overline{M}(\nu\widetilde{x})N} Q''$$

  All that then remains is to show $P' \mid (\!|\Psi|\!) \dot{\sim}_{\text{barb}} Q' \mid (\!|\Psi|\!)$. By Lemma 38 we know that

$$(\nu\widetilde{x})(P' \mid \overline{M_{P'}}\, N) \dot{\sim}_{\text{barb}} (\nu\widetilde{x})(Q'' \mid \overline{M_{P'}}\, N)$$

  and the desired conclusion follows by Lemma 42.

– *(extension of arbitrary assertion)* Immediate from the definition of $\mathcal{R}$ and Lemma 41.

Theorem 9. *In all observational psi-calculi, $P \dot{\sim}_{\text{barb}} Q$ iff $P \dot{\sim}_{\mathbf{1}} Q$.*

Proof. Immediate from Lemmas 39 and 43.