Published in Math. Struct. in Comp. Science, Volume 29, Special Issue 8 (A special issue on structural proof theory, automated reasoning and computation in celebration of Dale Miller's 60th birthday) September 2019, pp. 1275–1308. DOI: 10.1017/S0960129518000452

Constructing Weak Simulations from Linear Implications for Processes with Private Names

Ross Horne and Alwen Tiu

Received May 2017

This paper clarifies that linear implication defines a branching-time preorder, preserved in all contexts, when used to compare embeddings of process in non-commutative logic. The logic considered is a first-order extension of the proof system BV featuring a de Morgan dual pair of nominal quantifiers, called BV1. An embedding of π-calculus processes as formulae in BV1 is defined, and the soundness of linear implication in BV1 with respect to a notion of weak simulation in the π-calculus is established. A novel contribution of this work is that we generalise the notion of a "left proof" to a class of formulae sufficiently large to compare embeddings of processes, from which simulating execution steps are extracted. We illustrate the expressive power of BV1, by demonstrating results extend to the internal π-calculus, where privacy of inputs is guaranteed. We also remark that linear implication is strictly finer than any interleaving preorder.

1. Introduction

This paper contributes to a line of work formally relating logic and process calculi. The main interest is formally relating implication in a logical system to preorders over processes. In early work, Miller (1993), this is done by embedding processes as formulae in a fragment of linear logic, formulated in a sequent calculus, and by interpreting implication as a form of may testing preorder. We continue this investigation, but using a more general proof system based on the calculus of structures, Brünnler and Tiu (2001); Guglielmi and Straßburger (2001). The calculus of structures is a generalisation of the sequent calculus in which proof systems can be designed that cannot be expressed in the sequent calculus, Tiu (2006), notably the non-commutative logic BV, Guglielmi (2007), and its extensions NEL, Guglielmi and Straßburger (2011); Straßburger and Guglielmi (2011), and MAV, Horne (2015).

An established result, Bruscoli (2002), is that, for an embedding of processes in a fragment of CCS, Milner (1989), as formulae in BV, linear implication is strictly finer than (completed) trace inclusion. In other work, Horne et al. (2017), a tighter result is established, showing linear implication is strictly finer than pomset ideal inclusion — a classic notion of refinement for truly concurrent processes, Gischer (1988). Pomset ideals are defined with respect to certain homomorphisms over pomsets, which have finer non-interleaving properties than traces. Indeed BV was motivated by pomset logic, Retoré (1997), so formal links with pomsets should be no surprise.

This paper sharpens previous work on processes as formulae in two directions: firstly, we extend our embedding to more expressive process languages including the π-calculus, Milner et al. (1992); and, secondly, we establish the soundness of linear implication with respect to finer pro-

cess preorders, including weak simulation. A conference paper, Horne et al. (2016), introduced a first-order extension of MAV, called MAV1, featuring additive operators modelling choice, first-order universal and existential quantifiers and a novel de Morgan dual pair of nominal quantifiers called "new" and "wen", denoted $И$ and $Э$ respectively.[†] For the sake of clarity, here we restrict ourselves to the system BV1, excluding the additive operators that model choice in MAV1. This work clarifies that, for embeddings of processes as formulae, linear implication in BV1 is sound with respect to a fine notion of weak simulation called *complete weak open simulation*, in both the π-calculus and the πI-calculus, Sangiorgi (1996a). Some general proof normalisation techniques for extracting labelled transitions from proofs of certain forms are developed, that should be applicable to many process calculi beyond the fragment of the π-calculus explicitly considered in this paper.

Summary. Section 2 recalls established results for BV and lays out a roadmap for embeddings of processes as formulae. Section 3 introduces proof system BV1 and an embedding of π-calculus processes as formulae in BV1, along with statement of the main soundness result. Section 4, contains technical proof normalisation results, generalising established results on "left proofs" developed in related work. Section 5 combines lemmas regarding proof normalisation to prove the soundness of linear implication with respect to complete weak open simulation. Section 6 highlights that techniques extended to the private inputs in the πI-calculus.

2. A Roadmap for the Processes-as-Formulae Paradigm

This section touches on the history behind the search for a logical embedding of processes. We explain in what sense the calculus of structures addresses limitations of previous embeddings. The aim of this discussion is to clarify where the current paper sits in a roadmap towards a purely logical explanation of processes in the processes-as-formulae paradigm.

2.1. *Distinct logical approaches to the semantics of processes.*

Since the early days of linear logic, Girard (1987), applications of linear logic to modelling interactive concurrent systems have been suggested. Approaches can be classified along two major thrusts: proofs-as-process, and processes-as-formulae.

The *proofs-as-processes* paradigm Abramsky (1994); Bellin and Scott (1994) is inspired by the famous Curry-Howard correspondence between intuitionistic logic and typed λ-calculi. Recent work in the proofs-as-processes approach links formulae in linear logic with session types and proofs of the formulae with processes that inhabit the given session type, Caires et al. (2016). We do **not** follow the proofs-as-processes paradigm in this work.

In the approach we follow, the *processes-as-formulae* paradigm, processes are directly embedded as formulae. An early attempt at an embedding of process as formulae, Miller (1993), embeds the π-calculus as a theory in linear logic. In Miller's encoding, input and output prefixes are encoded via higher-order predicates, whose behaviour is defined via a theory in a higher-order fragment of linear logic. In that work, the semantics of a process is defined, in terms of

[†] Details of cut elimination for MAV1 appear in a supporting technical report, Horne et al. (2018).

the set of formulae provable in linear logic. Miller's approach falls short of a full processes-as-formulae embedding in the sense that input and output prefixes are not logical in nature. Also, name restriction is not handled.

Another related, but distinct, line of work is a deep embedding approach where process expressions are encoded as terms in a suitably expressive logic, and both the operational semantics and the process preorder or equivalence are encoded as (co-)inductive definitions McDowell et al. (2003); Tiu and Miller (2010); Bengtson and Parrow (2009). A limitation of such deep embeddings is implication in the logic does not directly define a preorder on processes.

2.2. *A purely logical approach to prefixes.*

The calculus of structures, Guglielmi (2007), revives research into the processes-as-formulae paradigm. The calculus of structures is sufficiently expressive to define extensions of linear logic with a non-commutative operator. As observed previously, Bruscoli (2002), for a fragment of CCS, a suitable non-commutative operator eliminates the need to use higher-order predicates to encode prefixes, as in the work, Miller (1993), discussed in the sub-section immediately above.

By defining a suitable logical system in the calculus of structures, we can directly map processes to formulae in that logical system. Furthermore, unlike approaches previously discussed, embedding of processes can be compared directly by using linear implication. The existence of a purely logical preorder over processes gives rise to natural questions:

— Where is linear implication situated in the spectrum of process preorders?
— Can expressive process calculi be embedded in systems defined in the calculus of structures?
— Given that there is a strong objective justification for this process preorder (cut elimination); are there also compelling applications for this process preorder in computer science?

The above questions set a broad agenda, to which the current paper contributes.

Figure 1 elaborates on the first question above. In Fig. 1, process preorders are divided along two axes: the linear-time/branching-time axis, van Glabbeek (1990), and the interleaving/causality-

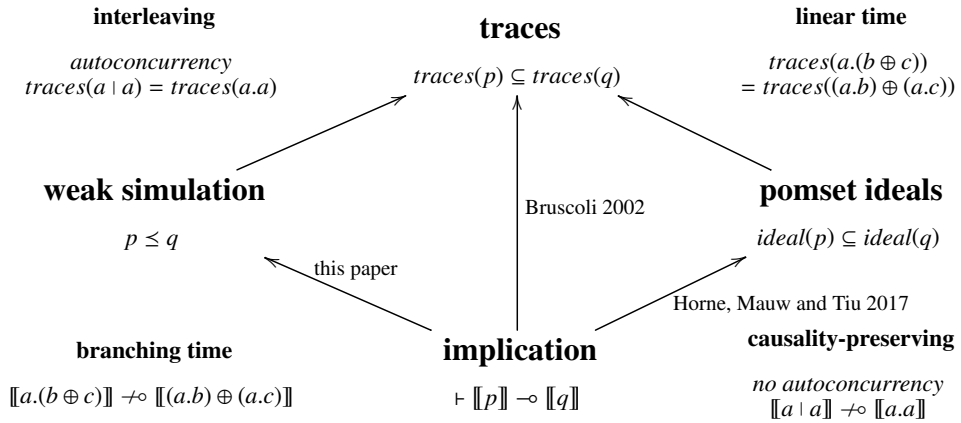

Fig. 1. A roadmap situating implication in the spectrum of process preorders. Arrows indicate strict soundness results. Linear implication distinguishes the most processes; while trace inclusion identifies the most processes. Weak simulation and pomset ideals are unrelated to each other.

preserving axis, Sassone et al. (1996). At the top of Fig. 1 is trace inclusion, defined by subset inclusion over the set of all traces of a process. Trace inclusion is widely considered to be the coarsest preorder over processes; hence as a minimal requirement all other preorders should be sound with respect to trace inclusion, as indicated by the arrows in the figure. Along the linear-time/branching-time axis, trace inclusion can be refined by various weak simulations which have finer properties regarding the distributivity of non-deterministic choice, indicated by $\oplus$ in this work. In the other direction, along the interleaving/causality-preserving axis, models such as pomset ideals, Gischer (1988), retain the causal relationships between events. Non-interleaving semantics ensure, for example, that, unlike trace inclusion and weak simulation, $a \triangleleft a$ and $a \parr a$ do not coincide — a property referred to as *autoconcurrency*, van Glabbeek and Goltz (2001).

We observe that linear implication has both branching-time and causality-preserving properties; hence is situated at the bottom of Fig. 1. This work formally establishes the soundness of linear implication with respect weak simulation, for a fragment the π-calculus.

2.3. *Recalling established results for a purely logical embedding of processes*

The first paper linking the calculus of structures with process calculi, Bruscoli (2002), embeds a fragment of CCS processes into the system BV in the calculus of structures. Formula P is *provable* in BV, written $\vdash P$, whenever we have a derivation of zero or more rule instances with conclusion P and premiss $\circ$ according the rules of BV in Fig. 2.

BCCS is a fragment of CCS, consisting of only parallel composition and input/output prefixes. The embedding of BCCS processes as formulae in BV is defined such that:

$$[\![1]\!]_{\mathsf{B}} = \circ \qquad [\![p \mid q]\!]_{\mathsf{B}} = [\![p]\!]_{\mathsf{B}} \parr [\![q]\!]_{\mathsf{B}} \qquad [\![a.p]\!]_{\mathsf{B}} = a \triangleleft [\![p]\!]_{\mathsf{B}} \qquad [\![\overline{a}.p]\!]_{\mathsf{B}} = \overline{a} \triangleleft [\![p]\!]_{\mathsf{B}}$$

A completed trace is defined by grammar $T ::= \circ \mid a \triangleleft T \mid \overline{a} \triangleleft T$. A process p_0 has completed trace $a_1 \triangleleft a_2 \ldots \triangleleft a_n$ whenever there exists a series of zero or more labelled transitions such that $p_0 \xrightarrow{a_1} p_1$, and $p_i \xrightarrow{a_{i+1}} p_{i+1}$ (skipping internal transitions due to interactions) such that p_n has no further actions to execute, hence is completed. The following theorem is established connecting completed traces with BCCS processes by using provability.

Theorem 2.1 (Bruscoli 2002). If T is a completed trace and p is a process in BCCS, then $\vdash T \multimap [\![p]\!]_{\mathsf{B}}$ in BV if and only if p has completed trace T.

We use negation-normal-forms where *linear implication* and *linear negation* are not primitive formulae, but instead are defined as functions that push negations to the atoms. Linear implica-

$$\begin{array}{lll} P ::= \circ & \text{(unit)} & \dfrac{C\{\circ\}}{C\{\overline{\alpha} \parr \alpha\}}\ \text{(atomic interaction)} \qquad \dfrac{C\{P \otimes (Q \parr R)\}}{C\{(P \otimes Q) \parr R\}}\ \text{(switch)} \\ \quad \alpha & \text{(atom)} & \\ \quad \overline{\alpha} & \text{(co-atom)} & \dfrac{C\{(P \parr R) \triangleleft (Q \parr S)\}}{C\{(P \triangleleft Q) \parr (R \triangleleft S)\}}\ \text{(sequence)} \\ \quad P \parr P & \text{(par)} & \\ \quad P \otimes P & \text{(times)} & \\ \quad P \triangleleft P & \text{(seq)} & (P, \parr, \circ) \text{ and } (P, \otimes, \circ) \text{ are commutative monoids, and } (P, \triangleleft, \circ) \text{ is a monoid.} \end{array}$$

Fig. 2. The syntax, inference rules and structural congruence for BV.

tion, $P \multimap Q$, is defined as $\overline{P} \parr Q$, where $\overline{P}$ is the linear negation of P, defined by the following function over formulae.

$$\overline{\overline{\alpha}} = \alpha \qquad \overline{P \otimes Q} = \overline{P} \parr \overline{Q} \qquad \overline{P \parr Q} = \overline{P} \otimes \overline{Q} \qquad \overline{\circ} = \circ \qquad \overline{P \triangleleft Q} = \overline{P} \triangleleft \overline{Q}$$

Linear negation defines de Morgan dualities. As in linear logic, the multiplicatives $\otimes$ and $\parr$ are de Morgan dual. *Seq* and the unit are self-dual, in the sense that their de Morgan dual operators are themselves. Cut elimination in BV can be formulated as follows.

Theorem 2.2 (Guglielmi 2007). If $\vdash C\{ P \otimes \overline{P} \}$ then $\vdash C\{ \circ \}$ in system BV.

Combining Theorem 2.2 and Theorem 2.1, we can establish the soundness of linear implication with respect to completed trace inclusion.

Corollary 2.3 (soundness). For BCCS processes p and q, if $\vdash [\![p]\!]_{\mathsf{B}} \multimap [\![q]\!]_{\mathsf{B}}$ then, for all completed traces T, if p has completed trace T then q has completed trace T.

Proof. Assume $\vdash [\![p]\!]_{\mathsf{B}} \multimap [\![q]\!]_{\mathsf{B}}$ holds and p has completed trace T. By Theorem 2.1, we have that $\vdash T \multimap [\![p]\!]_{\mathsf{B}}$ holds. Hence by Theorem 2.2, $\vdash T \multimap [\![q]\!]_{\mathsf{B}}$ holds. Thereby, by Theorem 2.1, process q has completed trace T, as required. □

The original paper on BV and BCCS, Bruscoli (2002), did not explicitly state the above corollary. That paper concerned only executions rather than the more subtle semantic issues surrounding process preorders used for refining processes, as explored in this work. As observed in Fig. 1, completeness of linear implication with respect to completed trace inclusion is impossible. The fact that linear implication does not exhibit autoconcurrency ($a \parr a \multimap a \triangleleft a$) is the simplest counterexample. Having no autoconcurrency is important for true-concurrency. For example, interleaving semantics, such as trace inclusion, exhibit autoconcurrency and therefore do not respect real-time. Hence linear implication is strictly finer than completed trace inclusion. Related work, Horne et al. (2017), tightens soundness by showing linear implication is strictly finer than pomset ideals, Gischer (1988), which are finer than completed traces. That work also makes explicit how timing properties are respected by linear implication.

The rest of this paper is dedicated to both: extending the discussion to the π-calculus, which demands a treatment of name binding; and also, tightening Fig. 1 by formally establishing the soundness of linear implication with respect to notions of weak simulation.

3. The First-order System BV1 and Embeddings of Mobile Processes

We recall a first-order extension of BV, called BV1, Horne et al. (2016), featuring a novel de Morgan dual pair of nominal quantifiers "new" and "wen" introduced to model private names as featured in the π-calculus. This section summarises key results required to recommend BV1 as a logical system, and provides an embedding of a fragment of the π-calculus in BV1. This enables us to state the main result of the paper — that linear implication is sound with respect to weak simulation — although key lemmas are postponed until later sections.

3.1. *The syntax, inference rules and structural rules of* BV1

The proof system BV1 extends BV with the first-order quantifiers *for all*, *new*, *wen* and *exists*. The syntax and structural congruence is presented in Fig. 3. The rules of the system, expressed as inference rules in Fig 4, can be applied in any context, where a context is a formula with a hole of the form $C\{\ \cdot\ \} ::= \{\ \cdot\ \} \mid C\{\ \cdot\ \} \odot P \mid P \odot C\{\ \cdot\ \} \mid \mho x.C\{\ \cdot\ \}$, where $\odot \in \{\triangleleft, \parr, \otimes\}$ and $\mho \in \{\exists, \forall, И, Э\}$ and $\{\ \cdot\ \}$ is a hole into which any formula can be plugged. We assume that quantifiers bind tighter than binary operators, e.g. $Иx.P \parr Q$ denotes $(Иx.P) \parr Q$.

structural congruence:

$Иx.Иy.P \equiv Иy.Иx.P \qquad Эx.Эy.P \equiv Эy.Эx.P$ (equivariance)

$(P, \parr, \circ)$ and $(P, \otimes, \circ)$ are commutative monoids, and $(P, \triangleleft, \circ)$ is a monoid.

linear negation:

$\overline{\overline{\alpha}} = \alpha \qquad \overline{P \otimes Q} = \overline{P} \parr \overline{Q} \qquad \overline{P \parr Q} = \overline{P} \otimes \overline{Q} \qquad \overline{\circ} = \circ \qquad \overline{P \triangleleft Q} = \overline{P} \triangleleft \overline{Q}$

$\overline{\forall x.P} = \exists x.\overline{P} \qquad \overline{\exists x.P} = \forall x.\overline{P} \qquad \overline{Иx.P} = Эx.\overline{P} \qquad \overline{Эx.P} = Иx.\overline{P}$

linear implication: $P \multimap Q = \overline{P} \parr Q$

$P ::= \circ$ (unit)
α (atom)
$\overline{\alpha}$ (co-atom)
$\forall x.P$ (for all)
$\exists x.P$ (exists)
$Иx.P$ (new)
$Эx.P$ (wen)
$P \parr P$ (par)
$P \otimes P$ (times)
$P \triangleleft P$ (seq)

Fig. 3. The syntax of BV1, structural congruence (including α-conversion for quantifiers), and definitions of linear negation and linear implication.

The structural congruence extends the structural congruence for BV with α-conversion for all quantifiers and *equivariance* for nominal quantifiers. Equivariance allows both nested *new* and *wen* operators to be exchanged. Note, for *exists* and *for all*, equivariance is a derived property; but equivariance must be explicitly induced in the structural congruence for nominal quantifiers.

Freshness is defined such that x is fresh for a formula P, written $x \# P$, if and only if x is not a member of the set of free variables of P, such that all quantifiers bind variables in their scope. A *substitution* is a mapping from variables to terms, which is an identity except for a finitely many variables. In this work, since we consider only name-passing calculi, we assume that the only terms in our logic are variables, but it is straightforward to extend the logic to allow function symbols as well. As standard, we assume that application of a substitution avoids capture of free variables. Application of substitutions is written in a postfix notation, e.g., $P\{^v/_x\}$ denotes an application of substitution $\{^v/_x\}$ to P.

Considerable creativity is permitted when defining the predicates and terms that form the atoms of the calculus. In work on session types atoms range over tuples containing any datatype equipped with a subtyping preorder, Ciobanu and Horne (2015). In work on attack trees, atoms are infinitely divisible as actions are refined, Horne et al. (2017). For the π-calculus embeddings in this paper, atoms are simply pairs of first order variables representing a channel and value passed on the given channel. Value passing extensions of the π-calculus can be embedded by extending with terms constructed from function symbols, constants and first-order variables.

Linear negation, defined by a function over formulae in Fig. 3, extends the de Morgan dualities for BV to quantifiers. The first-order quantifiers $\exists$ and $\forall$ are de Morgan dual, as are the nominal

$$\frac{C\{\ \forall x.(P \parr R)\ \}}{C\{\ \forall x.P \parr R\ \}}\ x \# R\ \text{(extrude1)} \qquad \frac{C\{\ \circ\ \}}{C\{\ \forall x.\circ\ \}}\ \text{(tidy1)}$$

$$\frac{C\{\ \forall x.P \triangleleft \forall x.S\ \}}{C\{\ \forall x.(P \triangleleft S)\ \}}\ \text{(medial1)} \qquad \frac{C\{\ P\{^{y}/_{x}\}\ \}}{C\{\ \exists x.P\ \}}\ \text{(select1)}$$

$$\frac{C\{\ Ии x.(P \parr Q)\ \}}{C\{\ Ии x.P \parr Э x.Q\ \}}\ \text{(close)} \qquad \frac{C\{\ Ии x.(P \parr R)\ \}}{C\{\ Ии x.P \parr R\ \}}\ x \# R\ \text{(extrude new)} \qquad \frac{C\{\ \circ\ \}}{C\{\ Ии x.\circ\ \}}\ \text{(tidy name)}$$

$$\frac{C\{\ Ии x.P\ \}}{C\{\ Э x.P\ \}}\ \text{(fresh)} \qquad \frac{C\{\ Э y.Ии x.P\ \}}{C\{\ Ии x.Э y.P\ \}}\ \text{(new wen)} \qquad \frac{C\{\ Ии y.\forall x.P\ \}}{C\{\ \forall x.Ии y.P\ \}}\ \text{(all new)} \qquad \frac{C\{\ Э y.\forall x.P\ \}}{C\{\ \forall x.Э y.P\ \}}\ \text{(all wen)}$$

$$\frac{C\{\ Ии x.P \triangleleft Ии x.S\ \}}{C\{\ Ии x.(P \triangleleft S)\ \}}\ \text{(medial new)} \qquad \frac{C\{\ Э x.(P \parr S)\ \}}{C\{\ Э x.P \parr Э x.S\ \}}\ \text{(suspend)} \qquad \frac{C\{\ Э x.(P \triangleleft S)\ \}}{C\{\ Э x.P \triangleleft Э x.S\ \}}\ \text{(suspend)}$$

$$\frac{C\{\ Э x.(P \parr R)\ \}}{C\{\ Э x.P \parr R\ \}}\ x \# R\ \text{(wen)} \qquad \frac{C\{\ Э x.(P \triangleleft R)\ \}}{C\{\ Э x.P \triangleleft R\ \}}\ x \# R\ \text{(wen)} \qquad \frac{C\{\ Э x.(R \triangleleft Q)\ \}}{C\{\ R \triangleleft Э x.Q\ \}}\ x \# R\ \text{(wen)}$$

Fig. 4. Inference rules, extending the three rules of BV, in Fig. 2, to formulae in system BV1.

quantifiers И and Э. Linear implication is a derived connective, and conservatively extends linear implication from the system BV to BV1.

The inference rules of BV1 are given in Fig. 4. A derivation is a sequence of zero or more inference rules, where the structural congruence can be applied at any point. We are particularly interested in proofs. Note we overload notation where $\overline{\overset{Q}{P}}$ means a derivation of any length with conclusion P and premiss Q, as opposed to the less compact notation for derivations $\overset{P}{\underset{Q}{\|}}$ employed in the literature. If it is important that a particular rule instance is applied, we make this clear in the surrounding text.

Definition 3.1. A *proof* in BV1 is a derivation of the form $\overline{\overset{\circ}{P}}$. When such a derivation exists, we say that P is provable, and write $\vdash P$.

Cut elimination holds for BV1 as a consequence of cut-elimination for MAV1, Horne et al. (2016). A full proof appears in a companion paper, Horne et al. (2018).

Theorem 3.2 (cut elimination). If $\vdash C\{\ P \otimes \overline{P}\ \}$ then $\vdash C\{\ \circ\ \}$ in system BV1.

Cut elimination is, of course, the corner stone of a proof system. Consistency for BV1 can be established immediately in the sense that for any provable formula with at least one atom, the linear negation is not provable. Of particular relevance to this work is the following corollary.

Corollary 3.3. Linear implication defines a preorder, a reflexive and transitive relation, preserved by all contexts.

3.2. *Embedding π-calculus processes as formulae in* BV1

We assume the reader is familiar with the syntax and operational semantics of the π-calculus, Milner et al. (1992). We consider a fragment of the π-calculus featuring parallel composition, input and output actions and private name binders. A (late) labelled transition system for this fragment of the π-calculus is recalled in Fig. 5. A small but notable syntactic departure from the literature is the use of action $\overline{x}[z]$ on labels to represent a bound output. This syntax is chosen to disambiguate semantically distinct concepts. In this paper we would like to discuss both the *input* of the π-calculus, and, later in this paper, the *private input* of the πI-calculus Sangiorgi (1996a). Traditionally, these inputs use the same syntax but have distinct semantics. To disambiguate which input we discuss we use $x(z)$ for the π-calculus input, which can receive both private and public names, and $x[z]$ for πI-calculus input, which can only receive a name if it is guaranteed to be private. The natural dual to private input is the output of a fresh private name, hence we also use syntax $\overline{x}[z]$ for private outputs in the π-calculus. In our embeddings, the intuitive duality between $x[z]$ and $\overline{x}[z]$ is put on a precise foundation by the use of the dual operators $\mathrel{⅋}$ and $\mathrel{⅄}$.

$$p ::= 1 \quad \text{(success)} \qquad \nu x.p \quad \text{(nu)} \qquad x(y).p \quad \text{(input)} \qquad \overline{x}y.p \quad \text{(output)} \qquad p \mid p \quad \text{(par)}$$

$$\pi ::= \tau \mid \overline{x}[z] \mid \overline{x}z \mid x(z) \quad \text{(actions)}$$

$$\frac{}{x(y).p \xrightarrow{x(y)} p} \qquad \frac{}{\overline{x}y.p \xrightarrow{\overline{x}y} p} \qquad \frac{p \xrightarrow{\overline{x}z} q}{\nu z.p \xrightarrow{\overline{x}[z]} q}\ x \neq z$$

$$\frac{p \xrightarrow{\pi} q}{\nu x.p \xrightarrow{\pi} \nu x.q}\ x \notin \mathrm{n}(\pi) \qquad \frac{p \xrightarrow{\pi} r}{p \mid q \xrightarrow{\pi} r \mid q}\ \text{if } \pi = \overline{x}[z] \text{ or } \pi = x(z),\ z \# q$$

$$\frac{p \xrightarrow{\overline{x}[z]} p' \quad q \xrightarrow{x(z)} q'}{p \mid q \xrightarrow{\tau} \nu z.(p' \mid q')} \qquad \frac{p \xrightarrow{\overline{x}y} p' \quad q \xrightarrow{x(z)} q'}{p \mid q \xrightarrow{\tau} p' \mid q'\{^{y}/_{z}\}}$$

Fig. 5. Syntax and labelled transitions for the π-calculus (plus symmetric rules for $p \mid q$). Function $n(.)$ is such that $n(x(y)) = n(\overline{x}[y]) = n(\overline{x}y) = \{x, y\}$ and $n(\tau) = \emptyset$. Freshness predicate $x \# p$ is such that x is fresh for p, where $z(x).p$ and $\nu x.p$ bind x in p.

Notice the constant 1 is used to indicate the successfully terminated process. We use the symbol 1, rather than 0 frequently used in the literature, so as to reserve 0 for representing deadlock. Note the use of 1 is standard for semantics which are sensitive to the difference between a process stopping because of deadlock and stopping because it has successfully fulfilled all its obligations and has not further pending actions to execute, e.g. Bernardi and Hennessy (2013).

In subsequent definitions, we require the following inductively defined predicate $\checkmark$, which holds whenever a process has successfully terminated.

Definition 3.4. $1\checkmark$ holds, and, if $p\checkmark$ and $q\checkmark$, then $(p \mid q)\checkmark$ holds and $(\nu x.p)\checkmark$ holds.

This work will prove linear implication is sound with respect to a notion of *weak simulation*. For weak simulations, if a process can perform a transition, then the simulating process can perform zero or more τ-transitions before matching the transition. In order to define weak simulation, we recall the following standard definitions.

Definition 3.5. $p_1 \Longrightarrow p_n$ whenever $p_1 = p_n$ or there exist processes $p_2, \ldots, p_{n-1}$ such that $p_i \xrightarrow{\tau} p_{i+1}$ for $1 \leq i < n$. If $p \Longrightarrow q$ and $q \xrightarrow{\pi} r$ and $r \Longrightarrow s$ then $p \overset{\pi}{\Longrightarrow} s$.

For a sharper result, we consider *complete weak open simulation*. For the *open* part of the definition, we require the following definition of a history, where a history is a neat representation of name distinctions in *open* simulation, Sangiorgi (1996b), employed in related work on deep embeddings of open bisimulation using ∇ quantifiers, Tiu and Miller (2010); Ahn et al. (2017).

Definition 3.6. A history is defined according to the grammar $h ::= \epsilon \mid h \cdot x^o \mid h \cdot x^i$. We denote with $n(h)$ the set of names (without the superscript o and i) appearing in h. Substitution σ respects history h whenever, for all h' and h'' such that $h = h' \cdot x^o \cdot h''$, we have $x\sigma = x$ and $x \notin n(h'\sigma)$.

Intuitively, the label i on a name x indicates that x is a name associated with an input action, and the label o indicates it is a name associated with an output action.

For *complete* simulations the termination potential is preserved. In this weak setting, preserving termination potential means, if a process has reached a successfully terminated state, as in Def. 3.4, then the simulating process can also perform zero or more internal τ-transitions to reach a successfully terminated state. This leads us to the following version of weak simulation.

Definition 3.7. A *complete weak open simulation* is a relation between processes, indexed by a history, such that, whenever $p\ \mathcal{R}^h\ q$ holds, all of the following hold:

— If substitution σ respects h, then $p\sigma\ \mathcal{R}^{h\sigma}\ q\sigma$ holds.
— If $p\checkmark$ then there exists q' such that $q \Longrightarrow q'$ and $q'\checkmark$.
— If $p \xrightarrow{\overline{x}z} p'$, there exists q' such that $q \overset{\overline{x}z}{\Longrightarrow} q'$ and $p'\ \mathcal{R}^h\ q'$.
— If $p \xrightarrow{\overline{x}[z]} p'$, where variable z is fresh for p, q and h, then there exists process q' such that $q \overset{\overline{x}[z]}{\Longrightarrow} q'$ and $p'\ \mathcal{R}^{h\cdot z^o}\ q'$.
— If $p \xrightarrow{x(z)} p'$, where variable z is fresh for p, q and h, then there exists process q' such that $q \overset{x(z)}{\Longrightarrow} q'$ and $p'\ \mathcal{R}^{h\cdot z^i}\ q'$.
— If $p \xrightarrow{\tau} p'$, then $p'\ \mathcal{R}^h\ q$.

p is *simulated* by q, written $p \preceq q$, whenever there exists a complete weak open simulation $\mathcal{R}$ such that $p\ \mathcal{R}^{x_0^i \cdot \cdots \cdot x_n^i}\ q$ where $\mathrm{fv}(p) \cup \mathrm{fv}(q) \subseteq \{x_0, \ldots, x_n\}$, where $\mathrm{fv}(p)$ denotes the set of free variables occurring in p.

We embed a fragment of the π-calculus in BV1. We assume an uninterpreted binary predicate *act* which we use to encode action prefixes representing a channel and a message transmitted on the channel. However, for a more compact presentation, we shall simply write xy, where x and y are names, to denote $act(x, y)$.

$$[\![1]\!]_\pi = \circ \qquad [\![p \mid q]\!]_\pi = [\![p]\!]_\pi \parr [\![q]\!]_\pi \qquad [\![\nu x.p]\!]_\pi = \mathit{И}x.[\![p]\!]_\pi$$

$$[\![x(z).p]\!]_\pi = \exists z.(xz \triangleleft [\![p]\!]_\pi) \qquad [\![\overline{x}z.p]\!]_\pi = \overline{xz} \triangleleft [\![p]\!]_\pi$$

The main result of this paper is linear implication is sound with respect to the above notion of weak simulation, and consequently all coarser preorders, including completed trace inclusion.

Theorem 3.8. For π-calculus processes p and q, if $\vdash [\![p]\!]_\pi \multimap [\![q]\!]_\pi$ in BV1, then $p \preceq q$.

The proof of the above theorem relies on Theorem 3.2 and results developed in the rest of this paper. We emphasise that, as for BV and BCCS, linear implication in BV1 is strictly finer than

complete weak open simulation, since Def. 3.7 is still an interleaving preorder and, unlike interleaving preorders, linear implication preserves causality (c.f. no autoconcurrency).

The following corollary of the above theorem emphasises a property of provable processes.

Corollary 3.9. If $\vdash [\![p]\!]_\pi$ in $\mathsf{BV1}$, then p can terminate successfully.

Proof. Assuming $\vdash [\![p]\!]_\pi$, we have $\vdash [\![1]\!]_\pi \multimap [\![p]\!]_\pi$ holds. By Theorem 3.8, we have $1 \preceq p$, hence since, $1\checkmark$, it must be the case that we have transitions $p \Longrightarrow p'$ such that $p'\checkmark$. □

The above property is sometimes referred to as a multiparty compatibility property, Deniélou and Yoshida (2013), since it indicates whether a collection of endpoints can be scheduled such that, collectively, they implement a protocol without deadlocking. This observation has been exploited to provide a purely logical explanation of multiparty compatibility in session types, based on provability, Ciobanu and Horne (2015). Multiparty compatibility tells us nothing about linear implication as a notion of process refinement, hence is a much weaker property than Theorem 3.8. The above corollary does however emphasises an advantage of respecting "complete" variants of weak simulations.

Further to the above benefit of complete simulations, observe that $\overline{x}x$ and $\overline{x}x \mid \overline{y}y$ are unrelated according to complete weak open simulation. In contrast, if instead consider only *weak open simulation*, dropping the *complete* condition regarding termination, then $\overline{x}x$ would be simulated by $\overline{x}x \mid \overline{y}y$ (a form of weakening property). Thus "complete" simulations, just as implication in linear logic, are resource sensitive — resources are preserved by refinement.

3.3. *Explanation for employing de Morgan dual nominal quantifiers*

We illustrate why neither universal quantification nor an established self-dual nominal quantifier, Pitts (2003); Miller and Tiu (2005); Gacek et al. (2011), are capable of soundly modelling name restriction in a processes-as-formulae embedding. In the following, observe $\nu x.(\overline{a}x \mid \overline{b}x)$ outputs a fresh name twice, once on channel a and once on channel b; but cannot output two distinct names in any execution. In contrast, observe $\nu x.\overline{a}x \mid \nu x.\overline{b}x$ outputs two distinct fresh names before terminating, but cannot output the same name twice in any execution. Processes $\nu x.(\overline{a}x \mid \overline{b}x)$ and $\nu x.\overline{a}x \mid \nu x.\overline{b}x$ must not be related by weak simulation.

Suppose that universal quantifiers were **unsoundly** used to encode private names. In this scenario, processes $\nu x.(\overline{a}x \mid \overline{b}x)$ and $\nu x.\overline{a}x \mid \nu x.\overline{b}x$ are (wrongly) encoded as formulae $\forall x.\big(\overline{ax} \parr \overline{bx}\big)$ and $\forall x.\overline{ax} \parr \forall x.\overline{bx}$ respectively. The implication $\forall x.\overline{ax} \parr \forall x.\overline{bx} \multimap \forall x.\big(\overline{ax} \parr \overline{bx}\big)$ is provable, but the respective processes are unrelated by simulation. To see why the processes are unrelated by simulation, observe both processes can output a fresh names on channel a as follows.

$$\nu x.(\overline{a}x \mid \overline{b}x) \xrightarrow{\overline{a}[x]} 1 \mid \overline{b}x \qquad \text{and} \qquad \nu x.\overline{a}x \mid \nu x.\overline{b}x \xrightarrow{\overline{a}[x]} 1 \mid \nu x.\overline{b}x$$

The above transitions lead to pair of processes $1 \mid \overline{b}x$ and $1 \mid \nu x.\overline{b}x$, where the former can only perform a free output transition $1 \mid \overline{b}x \xrightarrow{\overline{b}x} 1 \mid 1$, and the latter can only perform a bound output $1 \mid \nu x.\overline{b}x \xrightarrow{\overline{b}[x]} 1 \mid 1$. Hence neither process can simulate the other. Hence, under an encoding of private names using universal quantifiers, implication would not be sound with respect to weak simulation, and hence Theorem 3.8 would not hold under that encoding.

Additionally, an embedding of private names must also avoid the following *diagonalisation*

property: $\forall x.\forall y.P(x,y) \multimap \forall z.P(z,z)$. The self-dual nominal quantifiers of either Gabbay-Pitts or Miller-Tiu, as investigated in the calculus of structures, Roversi (2016), do successfully avoid the above diagonalisation property. However, encoding private names using any of these self-dual nominal quantifiers, say ∇, leads to the following problem. Suppose processes $\nu x.(\overline{a}x \mid \overline{b}x)$ and $\nu x.\overline{a}x \mid \nu x.\overline{b}x$ are encoded by formulae $\nabla x.(\overline{ax} \parr \overline{bx})$ and $\nabla x.\overline{ax} \parr \nabla x.\overline{bx}$ respectively. In this case, linear implication $\nabla x.(\overline{ax} \parr \overline{bx}) \multimap \nabla x.\overline{ax} \parr \nabla x.\overline{bx}$ is provable.[‡] This implication is also unsound with respect to simulation, since, as explained above, the processes are unrelated by simulation.

Our *new* quantifier И, distinct from the Gabbay-Pitts operator, addresses the above limitations of universal quantification and established self-dual nominal quantifiers. In addition to avoiding diagonalisation, our И quantifier does not distribute over parallel composition in either direction. The formulae $И x.(\overline{ax} \parr \overline{bx})$ and $И x.\overline{ax} \parr И x.\overline{bx}$ are, correctly, unrelated by linear implication.

3.4. *Translating from labelled transitions to proofs (but not the converse)*

Theorem 3.8 requires the following lemmas. The first follows by a trivial induction. The second translates any labelled transition into a proof and relies on cut elimination (Theorem 3.2).

Lemma 3.10. $[\![p]\!]_\pi\{^y/_x\} \equiv [\![p\{^y/_x\}]\!]_\pi$.

Lemma 3.11. The following statements hold:

— If $p \xrightarrow{\overline{x}[z]} q$ then $\vdash [\![\nu z.\overline{x}z.q]\!]_\pi \multimap [\![p]\!]_\pi$.
— If $p \xrightarrow{\overline{x}z} q$ then $\vdash [\![\overline{x}z.q]\!]_\pi \multimap [\![p]\!]_\pi$.
— If $p \xrightarrow{x(z)} q$ then $\vdash [\![x(z).q]\!]_\pi \multimap [\![p]\!]_\pi$.
— If $p \xrightarrow{\tau} q$ then $\vdash [\![q]\!]_\pi \multimap [\![p]\!]_\pi$.

Proof. The proof follows by structural induction on the derivation of a labelled transition.

Consider the base cases, where a label transition holds by an axiom of the labelled transition system. For axiom $x(z).p \xrightarrow{x(z)} p$, observe $\vdash [\![x(z).p]\!]_\pi \multimap [\![x(z).p]\!]_\pi$ holds by reflexivity of linear implication (Corollary 3.3). Similarly, for axiom $\overline{x}z.p \xrightarrow{\overline{x}z} p$, we have $\vdash [\![\overline{x}z.p]\!]_\pi \multimap [\![\overline{x}z.p]\!]_\pi$.

Consider the case of the open rule:

$$\frac{p \xrightarrow{\overline{x}z} q}{\nu z.p \xrightarrow{\overline{x}[z]} q}\ x \neq z$$

By assume as the induction hypothesis that $\vdash [\![\overline{x}z.q]\!]_\pi \multimap [\![p]\!]_\pi$ holds, such that $x \neq z$. Using this assumption, we can establish $\vdash [\![\overline{x}[z]\,.q]\!]_\pi \multimap [\![\nu z.p]\!]_\pi$ holds by the following proof, as required.

$$\dfrac{\dfrac{\dfrac{\circ}{И z.\circ}\ \text{by } \textit{tidy}}{И z.\Big(\big(xz \triangleleft \overline{[\![q]\!]_\pi}\big) \parr [\![p]\!]_\pi\Big)}\ \text{by the } \textit{induction hypothesis}}{\exists z.\Big(xz \triangleleft \overline{[\![q]\!]_\pi}\Big) \parr И z.[\![p]\!]_\pi}\ \text{by } \textit{close}$$

[‡] An extensive discussion on this implication appears in a companion paper, Horne et al. (2018).

Consider the case of the interaction rule involving a private name:

$$\frac{p \xrightarrow{x(z)} p' \quad q \xrightarrow{\overline{x}[z]} q'}{p \mid q \xrightarrow{\tau} \nu z.(p' \mid q')}$$

Assume as the induction hypotheses, $\vdash [\![x(z).p']\!]_\pi \multimap [\![p]\!]_\pi$ and $\vdash [\![\overline{x}[z].q']\!]_\pi \multimap [\![q]\!]_\pi$ hold. Now observe $\vdash ([\![x(z).p']\!]_\pi \multimap [\![p]\!]_\pi) \otimes ([\![\overline{x}[z].q']\!]_\pi \multimap [\![q]\!]_\pi) \multimap [\![\nu z.(p' \mid q')]\!]_\pi \multimap [\![p \mid q]\!]_\pi$ holds, by the following proof.

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{\circ}\ \text{tidy}}{\mathit{И}z.\circ}\ \text{reflexivity}}{\mathit{И}z.\big((xz \parr \overline{xz}) \triangleleft \big([\![p']\!]_\pi \parr [\![q']\!]_\pi \parr \big(\overline{[\![p']\!]_\pi} \otimes \overline{[\![q']\!]_\pi}\big)\big)\big)}\ \text{sequence}}{\mathit{И}z.\big((xz \triangleleft [\![p']\!]_\pi) \parr (\overline{xz} \triangleleft [\![q']\!]_\pi) \parr \big(\overline{[\![p']\!]_\pi} \otimes \overline{[\![q']\!]_\pi}\big)\big)}\ \text{select1}}{\mathit{И}z.\big(\exists z.(xz \triangleleft [\![p']\!]_\pi) \parr (\overline{xz} \triangleleft [\![q']\!]_\pi) \parr \big(\overline{[\![p']\!]_\pi} \otimes \overline{[\![q']\!]_\pi}\big)\big)}\ \text{extrude new}}{\exists z.(xz \triangleleft [\![p']\!]_\pi) \parr \mathit{И}z.\big((\overline{xz} \triangleleft [\![q']\!]_\pi) \parr \big(\overline{[\![p']\!]_\pi} \otimes \overline{[\![q']\!]_\pi}\big)\big)}\ \text{close}}{\exists z.(xz \triangleleft [\![p']\!]_\pi) \parr \mathit{И}z.(\overline{xz} \triangleleft [\![q']\!]_\pi) \parr \mathit{Э}z.\big(\overline{[\![p']\!]_\pi} \otimes \overline{[\![q']\!]_\pi}\big)}\ \text{reflexivity}}{\big(\big(\overline{[\![p]\!]_\pi} \parr [\![p]\!]_\pi\big) \otimes \exists z.(xz \triangleleft [\![p']\!]_\pi)\big) \parr \big(\big(\overline{[\![q]\!]_\pi} \parr [\![q]\!]_\pi\big) \otimes \mathit{И}z.(\overline{xz} \triangleleft [\![q']\!]_\pi)\big) \parr \mathit{Э}z.\big(\overline{[\![p']\!]_\pi} \otimes \overline{[\![q']\!]_\pi}\big)}\ \text{switch}}{\big(\exists z.(xz \triangleleft [\![p']\!]_\pi) \otimes \overline{[\![p]\!]_\pi}\big) \parr \big(\mathit{И}z.(\overline{xz} \triangleleft [\![q']\!]_\pi) \otimes \overline{[\![q]\!]_\pi}\big) \parr \big(\mathit{Э}z.\big(\overline{[\![p']\!]_\pi} \otimes \overline{[\![q']\!]_\pi}\big)\big) \parr [\![p]\!]_\pi \parr [\![q]\!]_\pi}$$

Hence by Theorem 3.2, $\vdash [\![\nu z.(p' \mid q')]\!]_\pi \multimap [\![p \mid q]\!]_\pi$, as required.

Consider the interaction rule of the following form, involving a free output:

$$\frac{p \xrightarrow{x(z)} p' \quad q \xrightarrow{\overline{x}y} q'}{p \mid q \xrightarrow{\tau} p'\{^y/_z\} \mid q'}$$

As the induction hypothesis assume that $\vdash [\![x(z).p']\!]_\pi \multimap [\![p]\!]_\pi$ and $\vdash [\![\overline{x}y.q']\!]_\pi \multimap [\![q]\!]_\pi$ hold. Also observe $\vdash ([\![x(z).p']\!]_\pi \multimap [\![p]\!]_\pi) \otimes ([\![\overline{x}y.q']\!]_\pi \multimap [\![q]\!]_\pi) \multimap [\![p'\{^y/_z\} \mid q']\!]_\pi \multimap [\![p \mid q]\!]_\pi$ holds, as established by the following proof and Lemma 3.10.

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{\circ}}{(xy \parr \overline{xy}) \triangleleft \big(\big(\overline{[\![q']\!]_\pi} \parr [\![q']\!]_\pi\big) \otimes \big(\overline{[\![p']\!]_\pi\{^y/_z\}} \parr [\![p']\!]_\pi\{^y/_z\}\big)\big)}\ \text{reflexivity}}{(xy \parr \overline{xy}) \triangleleft \big([\![q']\!]_\pi \parr [\![p']\!]_\pi\{^y/_z\} \parr \big(\overline{[\![p']\!]_\pi\{^y/_z\}} \otimes \overline{[\![q']\!]_\pi}\big)\big)}\ \text{switch}}{(xy \triangleleft [\![p']\!]_\pi\{^y/_z\}) \parr (\overline{xy} \triangleleft [\![q']\!]_\pi) \parr \big(\overline{[\![p']\!]_\pi\{^y/_z\}} \otimes \overline{[\![q']\!]_\pi}\big)}\ \text{sequence}}{\exists z.(xz \triangleleft [\![p']\!]_\pi) \parr (\overline{xy} \triangleleft [\![q']\!]_\pi) \parr \big(\overline{[\![p']\!]_\pi\{^y/_z\}} \otimes \overline{[\![q']\!]_\pi}\big)}\ \text{select1}}{\big(\big(\overline{[\![p]\!]_\pi} \parr [\![p]\!]_\pi\big) \otimes \exists z.(xz \triangleleft [\![p']\!]_\pi)\big) \parr ((\overline{[\![q]\!]_\pi} \parr [\![q]\!]_\pi) \otimes (\overline{xy} \triangleleft [\![q']\!]_\pi)) \parr \big(\overline{[\![p']\!]_\pi\{^y/_z\}} \otimes \overline{[\![q']\!]_\pi}\big)}\ \text{reflexivity}}{\big(\exists z.(xz \triangleleft [\![p']\!]_\pi) \otimes \overline{[\![p]\!]_\pi}\big) \parr \big((\overline{xy} \triangleleft [\![q']\!]_\pi) \otimes \overline{[\![q]\!]_\pi}\big) \parr \big(\overline{[\![p']\!]_\pi\{^y/_z\}} \otimes \overline{[\![q']\!]_\pi}\big) \parr [\![p]\!]_\pi \parr [\![q]\!]_\pi}\ \text{switch}$$

Hence, by Theorem 3.2, $\vdash [\![p'\{^y/_z\} \mid q']\!]_\pi \multimap [\![p \mid q]\!]_\pi$ holds, as required.

Consider the case of the contextual rule for new names instantiated for private output:

$$\frac{p \xrightarrow{\overline{x}[z]} q}{\nu y.p \xrightarrow{\overline{x}[z]} \nu y.q} \quad x \neq y \quad \text{and} \quad z \neq y$$

As the induction hypothesis, assume $\vdash [\![\overline{x}[z].q]\!]_\pi \multimap [\![p]\!]_\pi$ holds, where y is such that $x \neq y$ and

$z \neq y$. Using these assumptions, we can directly establish $\vdash [\![\overline{x}[z].\nu y.q]\!]_\pi \multimap [\![\nu y.p]\!]_\pi$ holds, by the following proof. Note the use of the *equivariance* structural rule.

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\circ}{И y.\circ}\ \text{tidy}}{И y.\left(\Im z.\left(xz \triangleleft \overline{[\![q]\!]_\pi}\right) \parr [\![p]\!]_\pi\right)}\ \text{induction hypothesis}}{\Im y.\Im z.\left(xz \triangleleft \overline{[\![q]\!]_\pi}\right) \parr И y.[\![p]\!]_\pi}\ \text{close}}{\Im z.\Im y.\left(xz \triangleleft \overline{[\![q]\!]_\pi}\right) \parr И y.[\![p]\!]_\pi}\ \text{equivariance}}{\Im z.\left(xz \triangleleft \Im y.\overline{[\![q]\!]_\pi}\right) \parr И y.[\![p]\!]_\pi}\ \text{wen, since } x \neq y \text{ and } z \neq y$$

Consider the contextual rule for parallel composition, instantiated for private output:

$$\dfrac{p \xrightarrow{\overline{x}[z]} q}{p \mid r \xrightarrow{\overline{x}[z]} q \mid r}\ z \# r$$

As the induction hypothesis assume $\vdash [\![\overline{x}[z].q]\!]_\pi \multimap [\![p]\!]_\pi$ holds, where $z \# r$. By these assumptions, $\vdash [\![\overline{x}[z].(q \mid r)]\!]_\pi \multimap [\![p \mid r]\!]_\pi$ holds, as established by the following proof.

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\circ}{\Im z.\left(xz \triangleleft \overline{[\![q]\!]_\pi}\right) \parr [\![p]\!]_\pi}\ \text{induction hypothesis}}{\Im z.\left(xz \triangleleft \left(\overline{[\![q]\!]_\pi} \otimes \left(\overline{[\![r]\!]_\pi} \parr [\![r]\!]_\pi\right)\right)\right) \parr [\![p]\!]_\pi}\ \text{reflexivity}}{\Im z.\left(xz \triangleleft \left(\left(\overline{[\![q]\!]_\pi} \otimes \overline{[\![r]\!]_\pi}\right) \parr [\![r]\!]_\pi\right)\right) \parr [\![p]\!]_\pi}\ \text{switch}}{\Im z.\left(\left(xz \triangleleft \left(\overline{[\![q]\!]_\pi} \otimes \overline{[\![r]\!]_\pi}\right)\right) \parr [\![r]\!]_\pi\right) \parr [\![p]\!]_\pi}\ \text{sequence}}{\Im z.\left(xz \triangleleft \left(\overline{[\![q]\!]_\pi} \otimes \overline{[\![r]\!]_\pi}\right)\right) \parr [\![p]\!]_\pi \parr [\![r]\!]_\pi}\ \text{wen, since } z \# r$$

Remaining cases for the contextual rules instantiated with free output, τ and input are similar to the previous two cases. Hence, by induction on the structure of the derivation of a labelled transition, a corresponding provable formula can be constructed as required. □

Remark 3.12. Interestingly, the *equivariance* structural rule (Fig. 3) is a design decision in the sense that cut elimination is still provable for BV1 without *equivariance*. However, *equivariance* is a requirement for soundly embedding the labelled transitions of the π-calculus (Lemma 3.11). Consider labelled transition $\nu y.\nu x.\overline{z}x.\overline{w}y \xrightarrow{\overline{z}[x]} \nu y.\overline{w}y$. The implication corresponding to the labelled transition: $\vdash [\![\nu x.\overline{z}x.\nu y.\overline{w}y]\!]_\pi \multimap [\![\nu y.\nu x.\overline{z}x.\overline{w}y]\!]_\pi$ is provable only if we have *equivariance*.

The converse of each statement in Lemma 3.11 does not necessarily hold. Hence, although we can embed labelled transitions as proofs, the same techniques cannot be applied to extract labelled transitions from proofs. Additional techniques developed in the next section are required.

4. Left Proofs and Normalisation Properties Required for Establishing the Main Result

This section provides technical devices required to establish the soundness of linear implication with respect to weak simulation (Theorem 3.8). These techniques, which are forms of *proof normalisation* properties, permit greater control of where in a proof the interaction of atoms,

instantiation of existential quantifiers, and extrusion of quantifiers each occur. These lemmas are critical for extracting labelled transitions from proofs in subsequent sections.

4.1. *Generalising established observations on left proofs in* BV

Before tackling BV1, we begin by generalising some established result for BV, Bruscoli (2002). The proof of Proposition 2.1 concerning BCCS employs a *normalisation* result transforming certain proofs in BV to "left proofs". In BV, a *left proof* is such that *atomic interaction* can only be applied inside *left contexts*, where the hole appears to the left of binary connectives, as follows: $\mathcal{L}\{\ \cdot\ \} ::= \{\cdot\} \mid \mathcal{L}\{\ \cdot\ \} \triangleleft P \mid \mathcal{L}\{\ \cdot\ \} \parr P \mid \mathcal{L}\{\ \cdot\ \} \otimes P$. For example, the formula $(a \triangleleft (b \parr c)) \parr \left(\left(\overline{a} \parr \overline{b}\right) \triangleleft \overline{c}\right)$ has a left proof where the *atomic interaction* rule must apply first to a before b, then finally c. The following established result is restricted to formulae in BV in which *times* $\otimes$ never appears.

Proposition 4.1 (Bruscoli 2002). If formula P contains no *times* operator and $\vdash P$ in BV, then there exists Q such that there is a derivation $\dfrac{Q}{P}$ using the *sequence* rule only and Q is provable using only the *atomic interaction* rule applied in left contexts.

The following example shows there are formulae for which there is no left proof.

$$\vdash \left(\left(\left(\overline{a} \triangleleft \overline{b}\right) \otimes \left(\overline{d} \triangleleft \overline{e}\right)\right) \triangleleft \left(\overline{c} \otimes \overline{f}\right)\right) \parr (a \triangleleft ((b \triangleleft c) \otimes (\overline{g} \parr g))) \parr \left(d \triangleleft \left((e \triangleleft f) \otimes \left(\overline{h} \parr h\right)\right)\right)$$

The above formula has no left proof in BV, where interaction is restricted to left contexts. The difficulty is that the atoms g and h must interact before *associativity* can be applied to allow the *sequence* rule to be correctly applied, as follows.

$$\dfrac{\dfrac{\dfrac{\circ}{\left(\left(\left(\overline{a} \triangleleft \overline{b}\right) \otimes \left(\overline{d} \triangleleft \overline{e}\right)\right) \triangleleft \left(\overline{c} \otimes \overline{f}\right)\right) \parr (((a \triangleleft b) \parr (d \triangleleft e)) \triangleleft (c \parr f))}\ \text{reflexivity}}{\left(\left(\left(\overline{a} \triangleleft \overline{b}\right) \otimes \left(\overline{d} \triangleleft \overline{e}\right)\right) \triangleleft \left(\overline{c} \otimes \overline{f}\right)\right) \parr (a \triangleleft b \triangleleft c) \parr (d \triangleleft e \triangleleft f)}\ \text{sequence}}{\left(\left(\left(\overline{a} \triangleleft \overline{b}\right) \otimes \left(\overline{d} \triangleleft \overline{e}\right)\right) \triangleleft \left(\overline{c} \otimes \overline{f}\right)\right) \parr (a \triangleleft ((b \triangleleft c) \otimes (\overline{g} \parr g))) \parr \left(d \triangleleft \left((e \triangleleft f) \otimes \left(\overline{h} \parr h\right)\right)\right)}\ \text{atomic interaction}$$

Fortunately, although there exist formulae with no left proof, it is possible to generalise Proposition 4.1 to formulae of the form $P \multimap Q$, where P and Q contain no *times* operator and *switch* may be applied along with *sequence*.

Proposition 4.2 (generalisation of Prop. 4.1). For formulae P and Q in which no *times* operator occurs. If $\vdash P \multimap Q$ in BV then then exists a left proof of $P \multimap Q$ in BV.

Now observe that embeddings of BCCS processes do not use the *times* operator. Hence, by the above observation, for any BCCS processes p and q, if $\vdash [\![p]\!]_{\mathsf{B}} \multimap [\![q]\!]_{\mathsf{B}}$ then we can always construct a left proof of the same formula. For example, $[\![b \mid c]\!]_{\mathsf{B}} \multimap \left[\!\left[\overline{a}.b.\overline{d} \mid a.c.d\right]\!\right]_{\mathsf{B}}$ has the

following left proof.

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\circ}{\overline{d} \parr d}\text{ atomic interaction}}{(\overline{c} \parr c) \triangleleft \left(\overline{d} \parr d\right)}\text{ atomic interaction}}{\left(\left(\overline{b} \parr b\right) \otimes (\overline{c} \parr c)\right) \triangleleft \left(\overline{d} \parr d\right)}\text{ atomic interaction}}{(\overline{a} \parr a) \triangleleft \left(\left(\overline{b} \parr b\right) \otimes (\overline{c} \parr c)\right) \triangleleft \left(\overline{d} \parr d\right)}\text{ atomic interaction}}{(\overline{a} \parr a) \triangleleft \left(b \parr c \parr \left(\overline{b} \otimes \overline{c}\right)\right) \triangleleft \left(\overline{d} \parr d\right)}\text{ switch}}{\left(\overline{b} \otimes \overline{c}\right) \parr \left(\overline{a} \triangleleft b \triangleleft \overline{d}\right) \parr (a \triangleleft c \triangleleft d)}\text{ sequence}$$

Furthermore, from a left proof, we know how to establish the soundness of linear implication with respect to *complete weak simulation* suitably defined for BCCS. For example, from the above proof, we can use a procedure to construct a complete weak simulation $\mathcal{S}$, defined to be the least relation containing the following pairs of processes:

$$(b \mid c)\ \mathcal{S}\ (\overline{a}.b.\overline{d} \mid a.c.d) \qquad (1 \mid c)\ \mathcal{S}\ (\overline{d} \mid c.d) \qquad (b \mid 1)\ \mathcal{S}\ (b.\overline{d} \mid d) \qquad (1 \mid 1)\ \mathcal{S}\ (\overline{d} \mid d)$$

In the interest of space, we proceed immediately to the π-calculus, rather than explaining this procedure on BCCS. The procedures for each calculus are similar, except the π-calculus requires additional mechanisms for handling quantifiers.

4.2. *Extending the concept of left proofs to BV1*

Here we define left proofs in BV1, by adapting the definition of left proofs for BV such that quantifiers are accommodated. The concept of a left context in BV1 is defined as follows.

Definition 4.3. A *left context* $\mathcal{L}\{\ \cdot\ \}$ is defined according to the following grammar:

$$\mathcal{L}\{\ \cdot\ \} ::= \quad \{\cdot\} \quad \mid \quad \mathcal{L}\{\ \cdot\ \} \triangleleft P \quad \mid \quad \mathcal{L}\{\ \cdot\ \} \parr P \quad \mid \quad \mathcal{L}\{\ \cdot\ \} \otimes P \quad \mid \quad \mathit{И}x.\mathcal{L}\{\ \cdot\ \} \quad \mid \quad \forall x.\mathcal{L}\{\ \cdot\ \}$$

The *interaction fragment* of BV1 consists of the rules *atomic interaction*, *tidy1* and *tidy name*. The *cooling fragment* consists of all other rules of BV1. A *left derivation* is a derivation where rules of the interaction fragment are applied only in a left context. A *left proof* of formula P in BV1 is a proof estalished using a left derivation.

We require a number of technical lemmas, to establish the key normalisation property, Proposition 4.11. Firstly, observe instances of *atomic interaction* can be pushed upwards in a proof until they are in a left context. In order to achieve this in BV1, we should first permute all *tidy* rules upwards, since they may obstruct an *atomic interaction* from being permuted upwards. To see

how this obstruction can happen, consider the following proof.

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\circ}{\left(\overline{ab} \parr ab\right)}\text{ atomic interaction}}{\left(\overline{ab} \parr ab\right) \triangleleft И z.\circ}\text{ tidy name}}{\left(\overline{ab} \parr ab\right) \triangleleft И z.(\overline{cz} \parr cz)}\text{ atomic interaction}}{\left(\overline{ab} \parr ab\right) \triangleleft И z.(\overline{cz} \parr \exists y.cy)}\text{ select1}}{\left(\overline{ab} \parr ab\right) \triangleleft (И z.\overline{cz} \parr \exists y.cy)}\text{ extrude new}}{\left(\overline{ab} \triangleleft И z.\overline{cz}\right) \parr (ab \triangleleft \exists y.cy)}\text{ sequence}}{\left(\overline{ab} \triangleleft И z.\overline{cz}\right) \parr \exists x.(ax \triangleleft \exists y.cy)}\text{ select1}$$

In the above proof, the *tidy name* rule followed by one of the *atomic interaction* rules can be permuted upwards until they are in a left context. The last three lines of the above proof can thereby be transformed to the following proof, where rules are only applied in left contexts.

$$\dfrac{\dfrac{\dfrac{\circ}{\nu z.\circ}\text{ tidy name}}{И z.(\overline{cz} \parr cz)}\text{ atomic interaction}}{\left(\overline{ab} \parr ab\right) \triangleleft И z.(\overline{cz} \parr cz)}\text{ atomic interaction}$$

There is a critical subtlety: a restriction is, as in BV, rules of the interaction fragment do not permute over all other rules. In particular, we should take care about where the *times* operator appears in derivations. To see why, consider the following derivation involving *tidy name*, *medial new* and the *times* operator.

$$\dfrac{\dfrac{C\{\ И x.(P \triangleleft Q) \triangleleft И x.(R \triangleleft S)\ \}}{C\{\ И x.(P \triangleleft Q \triangleleft R \triangleleft S)\ \}}\ \textit{medial new}}{C\{\ И x.(P \triangleleft (И y.\circ \otimes (Q \triangleleft R)) \triangleleft S)\ \}}\ \textit{tidy name}$$

The *tidy name* rule in the above derivation cannot be permuted above the *medial new* rule in BV1. For formulae in BV1 without *times*, the following normalisation property holds.

Lemma 4.4. Assume P and R contain no *times* operator. If there is a derivation $\dfrac{R}{P}$ in BV1, then there exists Q such that: there is a derivation $\dfrac{Q}{P}$ in the *cooling fragment* of BV1; and a derivation $\dfrac{R}{Q}$ in the *interaction fragment* of BV1.

In contrast to the above, notice *times* may appear in the formula in the following.

Lemma 4.5. If P is provable in the *interaction fragment* of BV1, then P has a left proof in BV1.

A key case for the above lemma, known already for BV, is the following derivation.

$$\dfrac{\dfrac{C\{\ \circ\ \}}{C\{\ a \parr \overline{a}\ \}}\text{ atomic interaction}}{C\left\{\ a \parr \left(\overline{a} \triangleleft \left(\overline{b} \parr b\right)\right)\ \right\}}\text{ atomic interaction}$$

Observe, in the derivation above, the atoms b do not interact in a left context. To ensure *atomic interaction* is always applied in a left context, the above derivation can be transformed to the derivation below.

$$\dfrac{\dfrac{\dfrac{C\{\circ\}}{C\{b \parr \overline{b}\}}\text{ atomic interaction}}{C\{(a \parr \overline{a}) \triangleleft (\overline{b} \parr b)\}}\text{ atomic interaction}}{C\{a \parr (\overline{a} \triangleleft (\overline{b} \parr b))\}}\text{ sequence}$$

No further problematic cases are introduced by quantifiers.

As a technical device, we define a form of restricted context called a *killing context*, Chaudhuri et al. (2011), consisting of a hole surrounded by quantifiers ∀ and И.

Definition 4.6 (killing context). Grammar $\mathcal{K}\{\ \cdot\ \} ::= \{\cdot\} \mid \forall x.\mathcal{K}\{\ \cdot\ \} \mid И x.\mathcal{K}\{\ \cdot\ \}$ defines killing contexts.

Splitting, proven in a companion paper as the main technical lemma required for proving cut elimination, Horne et al. (2018), is formulated as follows for BV1. Traditionally, in the sequent calculus, any connective can be selected and a corresponding rule applied. Splitting normalises proofs to generalise this feature of sequents to the calculus of structures.

Lemma 4.7 (splitting). The following statements hold.

— For any atom α, if $\vdash \overline{\alpha} \parr Q$, there exists killing context $\mathcal{K}\{\ \cdot\ \}$ such that if x appears in $\mathcal{K}\{\ \cdot\ \}$ then $x \# \alpha$, and there exists a derivation $\dfrac{\mathcal{K}\{\ \alpha\ \}}{Q}$.

— For any atom α, if $\vdash \alpha \parr Q$, there exists killing context $\mathcal{K}\{\ \cdot\ \}$ such that if x appears in $\mathcal{K}\{\ \cdot\ \}$ then $x \# \alpha$, and there exists derivation $\dfrac{\mathcal{K}\{\ \overline{\alpha}\ \}}{Q}$.

— If $\vdash (P \otimes Q) \parr R$, then there exist formulae V and W such that $\vdash P \parr V$ and $\vdash Q \parr W$, and killing context $\mathcal{K}\{\ \cdot\ \}$ such that $\dfrac{\mathcal{K}\{\ V \parr W\ \}}{R}$ and if x appears in $\mathcal{K}\{\ \cdot\ \}$ then $x \# (P \otimes Q)$.

— If $\vdash (P \triangleleft Q) \parr R$, then there exist formulae V and W such that $\vdash P \parr V$ and $\vdash Q \parr W$, and killing context $\mathcal{K}\{\ \cdot\ \}$ such that $\dfrac{\mathcal{K}\{\ V \triangleleft W\ \}}{R}$ and if x appears in $\mathcal{K}\{\ \cdot\ \}$ then $x \# (P \triangleleft Q)$.

— If $\vdash И x.P \parr Q$, then there exist formulae V and W where $x \# V$ and $\vdash P \parr W$ and either $V = W$ or $V = Э x.W$, such that we have derivation $\dfrac{V}{Q}$.

— If $\vdash Э x.P \parr Q$, then there exist formulae V and W where $x \# V$ and $\vdash P \parr W$ and either $V = W$ or $V = И x.W$, such that we have derivation $\dfrac{V}{Q}$.

— If $\vdash \exists x.P \parr Q$, then there exist formulae V and value v such that $\vdash P\{^{v}/_{x}\} \parr V$, and killing context $\mathcal{K}\{\ \cdot\ \}$ such that $\dfrac{\mathcal{K}\{\ V\ \}}{Q}$ and if y appears in $\mathcal{K}\{\ \cdot\ \}$ then $y \# (\exists x.P)$.

— If $\vdash \forall x.P \parr Q$ then, for any term t, $\vdash P\{^{t}/_{x}\} \parr Q$.

Although splitting was developed to establishing cut elimination, the technique has proven to be useful for solving other problems. Essentially, splitting can be used to guide a proof that proceeds by induction over the structure of formulae. Such structural induction is trivial in the

sequent calculus, since rules in sequents can only are applied to the root connective of a formula in a sequent. Hence, splitting can be seen as recovering part of the power of the sequent calculus in the more expressive setting of the calculus of structures.

A derivation over contexts $\dfrac{C'\{\ \cdot\ \}}{C\{\ \cdot\ \}}$ means, for all formulae U, $\dfrac{C'\{\ U\ \}}{C\{\ U\ \}}$, i.e., no rule is applied to the formula inserted in the hole. The following lemma ensures rules of the interaction fragment can be confined to parts of formulae of a certain form.

Lemma 4.8. Assume, for $\odot \in \{\otimes, \triangleleft, \parr\}$, we have derivation $\dfrac{\mathcal{K}\{\ Q_0 \odot Q_1\ \}}{S}$ in the *interaction fragment*. There exist context $C\{\ \cdot\ \}$ and formulae S_0 and S_1, such that, in the *interaction fragment*, $\dfrac{Q_0}{S_0}$, and $\dfrac{Q_1}{S_1}$, and $\dfrac{\mathcal{K}\{\ \cdot\ \}}{C\{\ \cdot\ \}}$; and also $\dfrac{C\{\ S_0 \odot S_1\ \}}{S}$ in the *cooling fragment*.

Assume $\dfrac{\mathcal{K}\{\ \mathbb{O}x.Q\ \}}{S}$ in the *interaction fragment*, where $\mathbb{O} \in \{\forall, И, \ni, \exists\}$. There exist context $C\{\ \cdot\ \}$ and formula R such that both $\dfrac{\mathcal{K}\{\ \cdot\ \}}{C\{\ \cdot\ \}}$ and $\dfrac{Q}{R}$ in the *interaction fragment*; and $\dfrac{C\{\ \mathbb{O}x.R\ \}}{S}$ in the *cooling fragment*.

We also require the following technical lemmas, used to break down certain derivations.

Lemma 4.9. Assume $\dfrac{\mathcal{K}\{\ \cdot\ \}}{C\{\ \cdot\ \}}$ using only the *interaction fragment*, and $\mathcal{K}\{\ \cdot\ \}$ is a killing context. In this situation, there exists left context $\mathcal{L}\{\ \cdot\ \}$ such that we can construct a *left derivation* $\dfrac{\mathcal{L}\{\ \cdot\ \}}{C\{\ \cdot\ \}}$ and, also, derivation $\dfrac{\mathcal{K}\{\ \cdot\ \}}{\mathcal{L}\{\ \cdot\ \}}$ using only the *interaction fragment*.

To understand the above lemma, consider context $(\overline{a} \parr a) \triangleleft \{\ \cdot\ \} \triangleleft \left(\overline{b} \parr b\right)$. There is no way to transform this context to $\{\ \cdot\ \}$, by applying interaction rules only in left contexts. However, we can first apply *atomic interaction* in a left context $\dfrac{\{\ \cdot\ \} \triangleleft \left(\overline{b} \parr b\right)}{(\overline{a} \parr a) \triangleleft \{\ \cdot\ \} \triangleleft \left(\overline{b} \parr b\right)}$, observing we obtain a premiss forming another left context. Assuming a formula P with a left proof is plugged into the context, we can construct a left proof $\dfrac{P \triangleleft \left(\overline{b} \parr b\right)}{\dfrac{\circ}{\overline{b} \parr b}}$. Thereby interactions to the right of the hole in the context are suspended to the end of the proof.

The following property is a simple distributivity property satisfied by left contexts.

Lemma 4.10. $\dfrac{\mathcal{L}\{\ Q \parr P\ \}}{Q \parr \mathcal{L}\{\ P\ \}}$, using the *cooling fragment*, for all left contexts $\mathcal{L}\{\ \cdot\ \}$, such that for all variables y appearing as binders in $\mathcal{L}\{\ \cdot\ \}$ we have $y \# Q$.

Although we cannot construct a left proof for all formulae containing *times*, as discussed for BV, we can handle a large class of formulae. The class of formulae we are concerned with is those formulae of the form $\mathcal{K}\{\ P \multimap Q\ \}$, where P and Q contain no *times* operator and $\mathcal{K}\{\ \cdot\ \}$ is a killing context. Fortunately, this class includes all formulae required for comparing embeddings

of processes using linear implication. By combining the above technical lemmas, we obtain the following more general property transforming proofs into left proofs.

Proposition 4.11. Assume P and Q contain no *times* operator and $\mathcal{K}\{\ \cdot\ \}$ is a killing context. If $\mathcal{K}\{\ P \multimap Q\ \}$ has a proof in BV1, then $\mathcal{K}\{\ P \multimap Q\ \}$ has a left proof in BV1.

Proof. The trick in this proof is, since $\overline{P}$ contains no *par* operator, we can proceed by induction on the structure of $\overline{P}$ in $\vdash \overline{P} \parr Q$, applying splitting (Lemma 4.7) at each step. Also observe the killing context can be trivially removed by repeatedly applying splitting.

Consider the base case where $P = \circ$, and $\vdash \circ \parr Q$. Since Q contains no *times* operator, by Lemma 4.4, there exists R such that we have derivation in the *cooling fragment* $\dfrac{R}{Q}$ and a proof of R in the *interaction fragment*. Since the proof of R is in the *interaction fragment*, by Lemma 4.5, there is a left proof of R, and hence a left proof of $\circ \parr Q$.

In the base case for atoms, $\vdash a \parr Q$ where Q contains no *times*. By splitting (Lemma 4.7), for some killing context $\mathcal{K}^1\{\ \cdot\ \}$, such that if y appears in $\mathcal{K}^1\{\ \cdot\ \}$ then $y \# \alpha$, we have a derivation $\dfrac{\mathcal{K}^1\{\ \overline{\alpha}\ \}}{Q}$. Hence, by Lemma 4.4, for some $C^1\{\ \cdot\ \}$ we have $\dfrac{C^1\{\ \overline{\alpha}\ \}}{Q}$ in the *cooling fragment* and $\dfrac{\mathcal{K}^1\{\ \overline{\alpha}\ \}}{C^1\{\ \overline{\alpha}\ \}}$ in the *interaction fragment*. By Lemma 4.9, there exists left context $\mathcal{L}^1\{\ \cdot\ \}$ such that we have *left derivation* $\dfrac{\mathcal{L}^1\{\ \overline{\alpha}\ \}}{C^1\{\ \overline{\alpha}\ \}}$ and in the *interaction fragment* we have derivation $\dfrac{\mathcal{K}^1\{\ \overline{\alpha}\ \}}{\mathcal{L}^1\{\ \overline{\alpha}\ \}}$. Now observe $\mathcal{L}^1\{\ \alpha \parr \overline{\alpha}\ \}$ is provable in the *interaction fragment* as follows: $\dfrac{\dfrac{\overset{\circ}{\overline{\mathcal{K}^1\{\ \circ\ \}}}}{\mathcal{K}^1\{\ \alpha \parr \overline{\alpha}\ \}}}{\mathcal{L}^1\{\ \alpha \parr \overline{\alpha}\ \}}$. Therefore, by Lemma 4.5, there is a left proof of $\mathcal{L}^1\{\ \alpha \parr \overline{\alpha}\ \}$. By using the left proof of $\mathcal{L}^1\{\ \alpha \parr \overline{\alpha}\ \}$, a left proof of $\alpha \parr Q$ can be constructed as follows. Note that Lemma 4.10 can be applied, since for all y appearing in $\mathcal{K}^1\{\ \cdot\ \}$ we have $y \# \alpha$, and in the derivation from $\mathcal{L}^1\{\ \cdot\ \}$ to $\mathcal{K}^1\{\ \cdot\ \}$ only the *interaction fragment* is applied; hence for all y appearing as binders in $\mathcal{L}^1\{\ \cdot\ \}$, we have $y \# \alpha$.

$$\begin{array}{l}
\dfrac{\overset{\circ}{}}{\mathcal{L}^1\{\ \alpha \parr \overline{\alpha}\ \}} \text{ a } \textit{left proof} \text{ (Lemma 4.5)} \\
\dfrac{\mathcal{L}^1\{\ \alpha \parr \overline{\alpha}\ \}}{\alpha \parr \mathcal{L}^1\{\ \overline{\alpha}\ \}} \text{ in the } \textit{cooling fragment} \text{ (Lemma 4.10)} \\
\dfrac{\alpha \parr \mathcal{L}^1\{\ \overline{\alpha}\ \}}{\alpha \parr C^1\{\ \overline{\alpha}\ \}} \text{ in the } \textit{a left derivation} \text{ (Lemma 4.9)} \\
\dfrac{\alpha \parr C^1\{\ \overline{\alpha}\ \}}{\alpha \parr Q} \text{ in the } \textit{cooling fragment} \text{ (Lemma 4.4)}
\end{array}$$

Inductive case involving times. Consider the inductive case where $\vdash (P_0 \parr P_1) \multimap R$, where P_0, P_1 and R contain no *times* operator (hence $\overline{P_0}$ and $\overline{P_1}$ contain no *par* operator). Firstly apply splitting: By Lemma 4.7, there are Q_0 and Q_1 containing no *times* operator and killing context $\mathcal{K}^2\{\ \cdot\ \}$ such that there is a derivation $\dfrac{\mathcal{K}^2\{\ Q_0 \parr Q_1\ \}}{R}$ such that $\vdash P_0 \multimap Q_0$ and $\vdash P_1 \multimap Q_1$.

By Lemma 4.4, since R and $\mathcal{K}^2\{\ Q_0 \parr Q_1\ \}$ contain no *times*, there exists S such that there is a derivation in the *cooling fragment* $\dfrac{S}{R}$, and derivation in the *interaction fragment* $\dfrac{\mathcal{K}^2\{\ Q_0 \parr Q_1\ \}}{S}$. Furthermore, by Lemma 4.8, there exists context $C^2\{\ \cdot\ \}$ and processes S_0 and S_1, such that we

have two derivations using only the *interaction fragment* $\dfrac{Q_0}{S_0}$ and $\dfrac{Q_1}{S_1}$, and derivation over contexts $\dfrac{\mathcal{K}^2\{\ \cdot\ \}}{C^2\{\ \cdot\ \}}$; and also we have in the *cooling fragment* a derivation $\dfrac{C^2\{\ S_0 \parr S_1\ \}}{S}$. By Lemma 4.9, there exists left context $\mathcal{L}^2\{\ \cdot\ \}$ such that there exists a *left derivation* over contexts $\dfrac{\mathcal{L}^2\{\ \cdot\ \}}{C^2\{\ \cdot\ \}}$, and another derivation over contexts $\dfrac{\mathcal{K}^2\{\ \cdot\ \}}{\mathcal{L}^2\{\ \cdot\ \}}$ using only the *interaction fragment*.

Now, since $\vdash P_0 \multimap Q_0$ and $\dfrac{Q_0}{S_0}$ we have $\dfrac{P_0 \multimap Q_0}{P_0 \multimap S_0}$, hence $\vdash P_0 \multimap S_0$; and similarly $\vdash P_1 \multimap S_1$. Therefore, by the induction hypothesis (noting the size of P_0 and P_1 are strictly less then the size of $P_0 \parr P_1$), we have $\vdash P_0 \multimap S_0$ and $\vdash P_1 \multimap S_1$ must have left proofs. Furthermore, since $\dfrac{\mathcal{K}^2\{\circ\}}{\mathcal{L}^2\{\circ\}}$ in the *interaction fragment*, we can construct a proof of $\mathcal{L}^2\{\circ\}$ using the *interaction fragment* only; hence, by Lemma 4.5, there is a left proof of $\mathcal{L}^2\{\circ\}$.

Using the derivations constructed above, and applying Lemma 4.10, we can construct the following left proof.

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\circ}{\mathcal{L}^2\{\circ\}}\ \text{a } \textit{left derivation} \text{ (Lemma 4.5)}}{\mathcal{L}^2\{\ P_1 \multimap S_1\ \}}\ \text{a } \textit{left derivation} \text{ (by induction hypothesis)}}{\mathcal{L}^2\{\ (P_0 \multimap S_0) \otimes (P_1 \multimap S_1)\ \}}\ \text{a } \textit{left derivation} \text{ (by induction hypothesis)}}{\mathcal{L}^2\left\{\ \left(\overline{P_0} \otimes \overline{P_1}\right) \parr S_0 \parr S_1\ \right\}}\ \text{by the } \textit{switch} \text{ rule}}{\left(\overline{P_0} \otimes \overline{P_1}\right) \parr \mathcal{L}^2\{\ S_0 \parr S_1\ \}}\ \text{in the } \textit{cooling fragment} \text{ (Lemma 4.10)}}{\left(\overline{P_0} \otimes \overline{P_1}\right) \parr C^2\{\ S_0 \parr S_1\ \}}\ \text{a } \textit{left derivation} \text{ (Lemma 4.9)}}{\left(\overline{P_0} \otimes \overline{P_1}\right) \parr S}\ \text{in the } \textit{cooling fragment} \text{ (Lemma 4.8)}}{(P_0 \parr P_1) \multimap R}\ \text{in the } \textit{cooling fragment} \text{ (Lemma 4.4)}$$

Case involving new quantifier. Consider the inductive case where $\vdash \forall x.P \multimap R$, where P and R contain no *times* operator (hence $\exists x.\overline{P}$ contains no *par*). Since, $\vdash \exists x.\overline{P} \parr R$, by splitting (Lemma 4.7), there are formulae V and W containing no *times* operator, such that: $x \# V$ and $\vdash \overline{P} \parr W$, either $V = W$ or $V = \forall x.W$, and also we have derivation $\dfrac{V}{R}$. There are two cases to consider, both resulting in a left proof of $\forall x.P \multimap R$.

In the first case where $V = W$, since $\dfrac{W}{R}$ and $\vdash \overline{P} \parr W$, we have $\vdash \overline{P} \parr R$; hence, by the induction hypothesis (which can be applied since the size of P is strictly less then the size of $\forall x.P$), there must be a left proof of $\vdash P \multimap R$. By α-conversion, we can assume $x \# R$, hence we can construct the following left proof:

$$\dfrac{\dfrac{\dfrac{\dfrac{\circ}{\forall x.\circ}\ \text{by the } \textit{tidy} \text{ rule}}{\forall x.(P \multimap R)}\ \text{a } \textit{left proof} \text{ by the induction hypothesis}}{\forall x.\overline{P} \parr R}\ \text{by the } \textit{extrude new} \text{ rule}}{\forall x.P \multimap R}\ \text{by the } \textit{fresh} \text{ rule}$$

In the second case where $V = \forall x.W$, since $\dfrac{\forall x.W}{R}$, and R and $\forall x.W$ contain no *times* operator,

by Lemma 4.4, there exists S such that $\dfrac{S}{R}$ in the *cooling fragment* and $\dfrac{\mathit{И}x.W}{S}$ in the *interaction fragment*. By Lemma 4.8, since $\dfrac{\mathit{И}x.W}{S}$ in the *interaction fragment*, there exist context $C^3\{\ \cdot\ \}$ and formula U such that we have: $\dfrac{C^3\{\ \mathit{И}x.U\ \}}{S}$ in the *cooling fragment*, and also $\dfrac{\{\ \cdot\ \}}{C^3\{\ \cdot\ \}}$ and $\dfrac{W}{U}$ using the *interaction fragment* only. Hence, by Lemma 4.9, there exists left context $\mathcal{L}^3\{\ \cdot\ \}$ such that we can construct a *left derivation* over contexts $\dfrac{\mathcal{L}^3\{\ \cdot\ \}}{C^3\{\ \cdot\ \}}$, and derivation $\dfrac{\{\ \cdot\ \}}{\mathcal{L}^3\{\ \cdot\ \}}$ using the *interaction fragment* only. Furthermore, observe, $\dfrac{\circ}{\mathit{И}x.\circ}$ $\mathcal{L}^3\{\ \mathit{И}x.\circ\ \}$ using the *interaction fragment* only; hence, by Lemma 4.5, we have a left proof of $\mathcal{L}^3\{\ \mathit{И}x.\circ\ \}$. Since $\dfrac{W}{U}$, and $\vdash P \multimap W$, we have $\vdash P \multimap U$; hence, by the induction hypothesis, we have a left proof of $P \multimap U$. Thus we can construct the following left proof, as required.

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\circ}{\mathcal{L}^3\{\ \mathit{И}x.\circ\ \}}\ \text{a } \textit{left derivation} \text{ (by Lemma 4.5)}}{\mathcal{L}^3\{\ \mathit{И}x.(P \multimap U)\ \}}\ \text{a } \textit{left derivation} \text{ (by induction hypothesis)}}{\mathcal{L}^3\{\ \mathit{И}x.P \multimap \mathit{И}x.U\ \}}\ \text{by the } \textit{close} \text{ rule}}{\mathit{И}x.P \multimap \mathcal{L}^3\{\ \mathit{И}x.U\ \}}\ \text{in the } \textit{cooling fragment} \text{ (by Lemma 4.10)}}{\mathit{И}x.P \multimap C^3\{\ \mathit{И}x.U\ \}}\ \text{a } \textit{left derivation} \text{ (by Lemma 4.9)}}{\mathit{И}x.P \multimap S}\ \text{in the } \textit{cooling fragment} \text{ (by Lemma 4.8)}}{\mathit{И}x.P \multimap R}\ \text{in the } \textit{cooling fragment} \text{ (by Lemma 4.4)}$$

Notice the above cases are similar. Remaining inductive cases follow the same pattern. □

In a left proof, we can always identify the bottommost *atomic interaction* rule, as expressed in the following lemma.

Lemma 4.12. For any left proof of P in BV1 containing at least one instance of the *atomic interaction* rule, there exists left context $\mathcal{L}\{\ \cdot\ \}$ and atom α such that:

— There exists a derivation of the form $\dfrac{\mathcal{L}\{\ \overline{\alpha} \parr \alpha\ \}}{P}$ in the *cooling fragment* of BV1.
— There exists a left proof of $\mathcal{L}\{\ \circ\ \}$ in BV1.

The subtlety of the above lemma is that the bottommost rule of the interaction fragment may be a *tidy* rule. In this scenario, it is always possible to remove such *tidy* rules from the proof until the bottommost rule of the interaction fragment is an instance of *atomic interaction*.

4.3. *Permuting nominal quantifiers to expand their scope*

We require two technical lemmas for normalising proofs with name binders. Extrusion of names is when a private name is sent over the network and becomes known by another process that receives the message. A challenge when handling extrusion is that the scope of the binder for the name should be expanded such that the process receiving the name is in the scope of the name

binder. In order to handle the expansion of the scope of name binders, we require the following technical devices.

In what follows the quantifier we wish to expand we identify using a box. We require the concept of when quantifiers are *connected* through a derivation. To identify connected quantifiers, we ensure that, if a quantifier is in a box in the conclusion, then it is also in a box in the premiss; where the following rules are treated in a special way.

— The *medial new* rule duplicates boxed quantifiers: $\dfrac{C\left\{\ \boxed{\text{И}x.}\ P \lhd \boxed{\text{И}x.}\ Q\ \right\}}{C\left\{\ \boxed{\text{И}x.}\ (P \lhd Q)\ \right\}}$.

— Instances of *suspend* and *close* rules connect two boxed nominal quantifiers in the conclusion, as follows, where $\odot \in \{\parr, \lhd\}$. $\dfrac{C\left\{\ \boxed{\text{Э}x.}\ (P \odot Q)\ \right\}}{C\left\{\ \boxed{\text{Э}x.}\ P \odot \boxed{\text{Э}x.}\ Q\ \right\}}$ and $\dfrac{C\left\{\ \boxed{\text{И}x.}\ (P \parr Q)\ \right\}}{C\left\{\ \boxed{\text{И}x.}\ P \parr \boxed{\text{Э}x.}\ Q\ \right\}}$.

— Instances of *tidy name* and *fresh* end or begin paths: $\dfrac{C\{\ \circ\ \}}{C\left\{\ \boxed{\text{И}x.}\ \circ\ \right\}}$ or $\dfrac{C\left\{\ \boxed{\text{И}x.}\ P\ \right\}}{C\{\ \text{Э}x.P\ \}}$ or $\dfrac{C\{\ \text{И}x.P\ \}}{C\left\{\ \boxed{\text{Э}x.}\ \right\}}$.

A proof, containing boxed nominal quantifiers is *minimally connected*, whenever all boxes are connected in the proof, as described above, and no box can be removed without either violating connectedness or removing all boxes. To understand this concept consider the following example derivation, where boxed quantifiers trace the path of a *new* connective.

$$
\begin{array}{cl}
\left(\text{И}y.\ \boxed{\text{И}x.}\ \circ \lhd \text{И}y.\ \boxed{\text{И}x.}\ \overline{by}\right) \parr \text{Э}y.by & \\
\hline
\left(\text{И}y.\ \boxed{\text{И}x.}\ (\overline{ax} \parr ax) \lhd \text{И}y.\ \boxed{\text{И}x.}\ \overline{by}\right) \parr \text{Э}y.by & \text{atomic interaction} \\
\hline
\left(\text{И}y.\left(\ \boxed{\text{И}x.}\ \overline{ax} \parr \boxed{\text{Э}x.}\ ax\right) \lhd \text{И}y.\ \boxed{\text{И}x.}\ \overline{by}\right) \parr \text{Э}y.by & \text{close} \\
\hline
\left(\left(\text{И}y.\ \boxed{\text{И}x.}\ \overline{ax} \parr \boxed{\text{Э}x.}\ ax\right) \lhd \text{И}y.\ \boxed{\text{И}x.}\ \overline{by}\right) \parr \text{Э}y.by & \text{extrude new} \\
\hline
\left(\text{И}y.\ \boxed{\text{И}x.}\ \overline{ax} \lhd \text{И}y.\ \boxed{\text{И}x.}\ \overline{by}\right) \parr \boxed{\text{Э}x.}\ ax \parr \text{Э}y.by & \text{sequence} \\
\hline
\text{И}y.\left(\ \boxed{\text{И}x.}\ \overline{ax} \lhd \boxed{\text{И}x.}\ \overline{by}\right) \parr \boxed{\text{Э}x.}\ ax \parr \text{Э}y.by & \text{medial new} \\
\hline
\text{И}y.\ \boxed{\text{И}x.}\ \left(\overline{ax} \lhd \overline{by}\right) \parr \boxed{\text{Э}x.}\ ax \parr \text{Э}y.by & \text{medial new}
\end{array}
$$

We also require the following definition removing boxed quantifiers from formulae, where κ is $\circ$ or an atom, $\mho \in \{\forall, \text{И}, \text{Э}, \exists\}$ and $\odot \in \{\otimes, \lhd, \parr\}$.

$$\mathrm{rm}(\ \boxed{\mho x.}\ P) = \mathrm{rm}(P) \qquad \mathrm{rm}(\mho y.P) = \mho y.\mathrm{rm}(P) \qquad \mathrm{rm}(P \odot Q) = \mathrm{rm}(P) \odot \mathrm{rm}(Q) \qquad \mathrm{rm}(\kappa) = \kappa$$

Using the above devices we can establish the following normalisation lemma.

Lemma 4.13. If $\vdash C\{\ P\ \}$, where P contains at least one minimally connected boxed new quantifier $\boxed{\text{И}x.}$, and x only appears bound by boxed quantifiers in $C\{\ P\ \}$ (avoiding name confusion), then $\vdash C\{\ \text{И}x.\mathrm{rm}(P)\ \}$.

Proof. The proof is established by induction on the structure of the derivation. One case is illustrated. Consider the case where the bottommost rule of a proof is of the following form,

where $x \# P$, $x \# R$ and no boxed quantifier appears in P; hence $\mathrm{rm}(P) = P$.

$$\frac{C\{\ P \parr \boxed{И x.}\ (Q \parr R)\ \}}{C\{\ P \parr \boxed{И x.}\ Q \parr R\ \}}\ \text{extrude new}$$

By the induction hypothesis $C\{\ И x.\mathrm{rm}(P \parr Q \parr R)\ \}$ is provable, hence $\vdash C\{\ P \parr И x.\mathrm{rm}(Q \parr R)\ \}$, since $\dfrac{C\{\ И x.(P \parr \mathrm{rm}(Q \parr R))\ \}}{C\{\ P \parr И x.\mathrm{rm}(Q \parr R)\ \}}$ and $C\{\ И x.(P \parr \mathrm{rm}(Q \parr R))\ \} = C\{\ И x.\mathrm{rm}(P \parr Q \parr R)\ \}$. □

Consider again the example provided immediately before Lemma 4.13. The function rm can be applied to that example derivation to obtain the derivation below. Observe the boxed quantifiers and redundant rules have been deleted; also $И x$ is reinserted with a wider scope.

$$\dfrac{\dfrac{\dfrac{\dfrac{И x.\big(И y.\circ \triangleleft И y.\overline{by}\big) \parr Э y.by}{И x.\big(И y.(\overline{ax} \parr ax) \triangleleft И y.\overline{by}\big) \parr Э y.by}\ \text{atomic interaction}}{И x.\big((И y.\overline{ax} \parr ax) \triangleleft И y.\overline{by}\big) \parr Э y.by}\ \text{extrude new}}{И x.\big(\big(И y.\overline{ax} \triangleleft И y.\overline{by}\big) \parr ax\big) \parr Э y.by}\ \text{sequence}}{И x.\big(И y.\big(\overline{ax} \triangleleft \overline{by}\big) \parr ax\big) \parr Э y.by}\ \text{medial new}$$

Note the scope of universal quantifiers in the conclusion of a proof can also be expanded, due to the following implications, where $\odot \in \{\otimes, \triangleleft, \parr\}$, $Ọ \in \{\forall, И, Э, \exists\}$ and $x \# R$.

$$\vdash \forall x.P \odot R \multimap \forall x.(P \odot R) \qquad \vdash R \odot \forall x.Q \multimap \forall x.(R \odot Q) \qquad \vdash Ọ x.\forall y.P \multimap \forall y.Ọ x.P$$

The above implications show that universal quantification distributes over all other operators, as long as name capture is avoided. Thus the process of expanding the scope of universal quantifiers is simpler than for nominal quantifiers.

4.4. *Permuting exists to the bottom of certain derivations*

For certain derivations, it is possible to permute the *select1* rule, concerning the instantiation of existential quantifiers, to the bottom of the derivation; as long as the *select1* rule does not commute with an extrusion rule binding a variable in the term introduced. To understand this subtlety, consider the following derivation, where $x \# P$.

$$\dfrac{\dfrac{И x.\big(\overline{ax} \parr \big(ax \triangleleft P\{^{x}/_{y}\}\big)\big)}{И x.(\overline{ax} \parr \exists y.(ay \triangleleft P))}\ \text{select1}}{И x.\overline{ax} \parr \exists y.(ay \triangleleft P)}\ \text{extrude new}$$

The instance of the *select1* rule above cannot permute with the instance of the *extrude name* rule. Fortunately, this situation can be avoided for formulae relevant to embedding processes by first expanding the scope of nominal and universal quantifiers, using Lemma 4.13. This leads to the following normalisation property.

Lemma 4.14. Consider a derivation in the cooling fragment of the form $\dfrac{\mathcal{L}'\{\ \boxed{\exists x.}\ Q\ \}}{\mathcal{L}\{\ \boxed{\exists x.}\ P\ \}}$, where an existential quantifier is identified by a box. Also, assume t is such that: if there is a quantifier

$$\frac{C\Big\{\ \mathcal{O}y.\ \big(R \parr \exists x.\ S\big)\ \Big\}}{C\Big\{\ \mathcal{O}y.\ R \parr \exists x.\ S\ \Big\}}$$

$\mathcal{O} \in \{\forall, И\}$ such that a rule of of the form appears in the derivation, then the variable y bound by $\mathcal{O}$ is such that $y \# t$. In such a scenario, a derivation $\dfrac{\mathcal{L}'\{\ Q\{^{t}/_{x}\}\ \}}{\mathcal{L}\{\ P\{^{t}/_{x}\}\ \}}$ can be constructed in the cooling fragment.

5. Constructing Labelled Transitions from Proofs

This section, combines results from the previous sections to establish the main result of the paper (Theorem 3.8). In order to ensure all scenarios are covered, we require the following definition and lemma. Observe that process contexts transform to left contexts when used in embeddings.

Definition 5.1. A *process context* $\mathcal{P}\{\ \cdot\ \}$ is defined according to the following grammar, where p ranges over π-calculus processes: $\mathcal{P}\{\ \cdot\ \} ::= \{\cdot\} \quad | \quad \mathcal{P}\{\ \cdot\ \} \mid p \quad | \quad p \mid \mathcal{P}\{\ \cdot\ \} \quad | \quad \nu x.\mathcal{P}\{\ \cdot\ \}$.

We say a process context $\mathcal{P}\{\ \cdot\ \}$ binds variable x, when there appears a $\nu x.$ in scope of the hole of the context. That is the context is such that $\mathcal{P}\{\ \cdot\ \} = \mathcal{P}^0\Big\{\ \nu x.\mathcal{P}^1\{\ \cdot\ \}\ \Big\}$.

Lemma 5.2.

— For π-calculus process q and process context $\mathcal{P}\{\ \cdot\ \}$ that binds neither x nor z, the following labelled transition exists: $\mathcal{P}\{\ \overline{x}z.q\ \} \xrightarrow{\overline{x}z} \mathcal{P}\{\ q\ \}$.

— Assuming x and z are not bound by $\mathcal{P}^1\{\ \cdot\ \}$ and $\mathcal{P}^2\{\ \cdot\ \}$, there exists the following transition.

$$\mathcal{P}^0\Big\{\ \mathcal{P}^1\{\ \overline{x}z.p\ \} \mid \mathcal{P}^2\{\ x(y).q\ \}\ \Big\} \xrightarrow{\tau} \mathcal{P}^0\Big\{\ \mathcal{P}^1\{\ p\ \} \mid \mathcal{P}^2\Big\{\ q\{^{z}/_{y}\}\ \Big\}\ \Big\}$$

— Where x and z are not bound by $\mathcal{P}^1\{\ \cdot\ \}$, $\mathcal{P}^2\{\ \cdot\ \}$, or $\mathcal{P}^3\{\ \cdot\ \}$ and z is fresh for all processes appearing in the contexts $\mathcal{P}^1\{\ \cdot\ \}$ and $\mathcal{P}^3\{\ \cdot\ \}$, the following transition exists:

$$\mathcal{P}^0\Big\{\ \mathcal{P}^1\Big\{\ \nu z.\mathcal{P}^2\{\ \overline{x}z.p\ \}\ \Big\} \mid \mathcal{P}^3\{\ x(z).q\ \}\ \Big\} \xrightarrow{\tau} \mathcal{P}^0\Big\{\ \nu z.\Big(\mathcal{P}^1\Big\{\ \mathcal{P}^2\{\ p\ \}\ \Big\} \mid \mathcal{P}^3\{\ q\ \}\Big)\ \Big\}$$

— Assume q is a π-calculus process and $\mathcal{P}^0\{\ \cdot\ \}$ and $\mathcal{P}^1\{\ \cdot\ \}$ are a process contexts that do not bind x and z; and z is fresh for $\mathcal{P}^0\{\ \cdot\ \}$. Under these conditions, the following labelled transition exists: $\mathcal{P}^0\Big\{\ \nu z.\mathcal{P}^1\{\ \overline{x}z.q\ \}\ \Big\} \xrightarrow{\overline{x}[z]} \mathcal{P}^0\Big\{\ \mathcal{P}^1\{\ q\ \}\ \Big\}$.

— Assuming x and y are not bound by $\mathcal{P}\{\ \cdot\ \}$ and x is fresh for $\mathcal{P}\{\ \cdot\ \}$, there is always the following labelled transition: $\mathcal{P}\{\ y(x).q\ \} \xrightarrow{y(x)} \mathcal{P}\{\ q\ \}$.

Similar situations where processes composed by parallel composition are swapped also hold.

Proof. The most involved case where a bound output interacts with an input is presented. Consider where x and z are not bound by $\mathcal{P}^1\{\ \cdot\ \}$, $\mathcal{P}^2\{\ \cdot\ \}$, or $\mathcal{P}^3\{\ \cdot\ \}$ and z is fresh for all processes appearing in the contexts $\mathcal{P}^1\{\ \cdot\ \}$ and $\mathcal{P}^3\{\ \cdot\ \}$.

Since $x(z).q \xrightarrow{x(z)} q$ and $\overline{x}z.p \xrightarrow{\overline{x}z} p$, by induction on the structure of $\mathcal{P}^3\{\ \cdot\ \}$ and $\mathcal{P}^2\{\ \cdot\ \}$, we have $\mathcal{P}^3\{\ x(z).q\ \} \xrightarrow{x(z)} \mathcal{P}^3\{\ q\ \}$ and $\mathcal{P}^2\{\ \overline{x}z.p\ \} \xrightarrow{\overline{x}z} \mathcal{P}^2\{\ p\ \}$. Thereby $\nu z.\mathcal{P}^2\{\ \overline{x}z.p\ \} \xrightarrow{\overline{x}[z]} \mathcal{P}^2\{\ p\ \}$, so, by induction on the structure of $\mathcal{P}^1\{\ \cdot\ \}$, we have $\mathcal{P}^1\Big\{\ \nu z.\mathcal{P}^2\{\ \overline{x}z.p\ \}\ \Big\} \xrightarrow{\overline{x}[z]} \mathcal{P}^1\Big\{\ \mathcal{P}^2\{\ p\ \}\ \Big\}$. Thereby, by interaction of labels, $\mathcal{P}^1\Big\{\ \nu z.\mathcal{P}^2\{\ \overline{x}z.p\ \}\ \Big\} \mid \mathcal{P}^3\{\ x(z).q\ \} \xrightarrow{\tau} \nu z.\Big(\mathcal{P}^1\Big\{\ \mathcal{P}^2\{\ p\ \}\ \Big\} \mid \mathcal{P}^3\{\ q\ \}\Big)$. Hence, by structural induction on context $\mathcal{P}^0\{\ \cdot\ \}$, we have constructed the following transition as required: $\mathcal{P}^0\Big\{\ \mathcal{P}^1\Big\{\ \nu z.\mathcal{P}^2\{\ \overline{x}z.p\ \}\ \Big\} \mid \mathcal{P}^3\{\ x(z).q\ \}\ \Big\} \xrightarrow{\tau} \mathcal{P}^0\Big\{\ \nu z.\Big(\mathcal{P}^1\Big\{\ \mathcal{P}^2\{\ p\ \}\ \Big\} \mid \mathcal{P}^3\{\ q\ \}\Big)\ \Big\}$. □

5.1. *Key scenarios for extracting executions*

Here we identify key scenarios required for inducing labelled transitions, while preserving provability. In each scenario, consider a proof of the following form, where $\mathcal{L}\{\ \cdot\ \}$ is a left context.

$$\dfrac{\dfrac{\dfrac{\circ}{\mathcal{L}\{\circ\}}\ \text{a } \textit{left proof}}{\mathcal{L}\{\ \overline{xz} \parr xz\ \}}\ \text{using } \textit{atomic interaction}}{P}\ \text{using the } \textit{cooling fragment} \text{ only}$$

Recall, by Lemma 4.12, this form can always be achieved for left proofs with at least one atom.

Now, observe, moving upwards in a proof no rule of BV1 introduces atoms or quantifiers. Hence atom xz and co-atom $\overline{xz}$ involved in the atomic interaction rule must also appear in the conclusion, P. Furthermore, since the bottommost interaction rule is always in a left context and cooling rules preserve this property, the atom also appears in a left context in the conclusion of the proof. These observations allow us to identify the following scenarios, all essential for the main result of this paper.

5.1.1. *Scenario for inducing free output transitions.* The simplest scenario is where, in a left proof, the bottommost *atomic interaction* rule involves an atom xz that can be traced to two particular sub-formulae in the conclusion of the proof, each an embedding of a process $\overline{x}z.p$ and $\mathcal{P}\{\ \overline{x}z.q\ \}$ respectively.

Lemma 5.3. Assume $\vdash\ \mathcal{L}\{\circ\}$ holds and $\dfrac{\mathcal{L}\{\ xz \parr \overline{xz}\ \}}{\mathcal{K}\{\ [\![\overline{x}z.p]\!]_\pi \multimap [\![\mathcal{P}\{\ \overline{x}z.q\ \}]\!]_\pi\ \}}$ is a derivation in the cooling fragment of BV1, such that p and q are π-calculus processes, $\mathcal{K}\{\ \cdot\ \}$ is a killing context, $\mathcal{L}\{\ \cdot\ \}$ is a left context, $\mathcal{P}\{\ \cdot\ \}$ is a process context, x and z are not bound by $\mathcal{P}\{\ \cdot\ \}$, and the instances of xz in the conclusion and premiss are connected.

Under these assumptions, $\vdash \mathcal{K}\{\ [\![p]\!]_\pi \multimap [\![\mathcal{P}\{\ q\ \}]\!]_\pi\ \}$ holds.

Notice the former process $\overline{x}z.p$ is in a form where it has committed to performing a free output, indicating the process can only perform action $\overline{x}z$, after which process p is reached. The latter process $\mathcal{P}\{\ \overline{x}z.q\ \}$ is ready to perform an action $\overline{x}z$, assuming x and z are free variables. In this scenario, the action in each sub-formula can be removed from the proof while preserving provability, as described in what follows.

This scenario is no more complex than the proof of the established result relating executions of BCCS certain proofs in BV, Theorem 2.1. Simply remove the instances of atoms xz and $\overline{xz}$ from the derivation in the cooling fragment to obtain derivation $\dfrac{\mathcal{L}\{\circ\}}{\mathcal{K}\Big\{\ \overline{[\![p]\!]_\pi} \parr [\![\mathcal{P}\{\ q\ \}]\!]_\pi\ \Big\}}$, from which a proof of $\mathcal{K}\{\ [\![p]\!]_\pi \multimap [\![\mathcal{P}\{\ q\ \}]\!]_\pi\ \}$ can be constructed, as required.

Observe some rules become redundant, so are removed from the proof in this process. For

example consider a left proof beginning with the following derivation.

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\circ}{[\![q]\!]_\pi \multimap \mathit{И}y.[\![p]\!]_\pi}\ \text{a } \textit{left proof}}{\mathit{И}y.\circ \triangleleft ([\![q]\!]_\pi \multimap \mathit{И}y.[\![p]\!]_\pi)}\ \text{tidy new}}{\mathit{И}y.\big(\ xz\ \parr\ \overline{xz}\ \big) \triangleleft ([\![q]\!]_\pi \multimap \mathit{И}y.[\![p]\!]_\pi)}\ \text{atomic interaction}}{\big(\ xz\ \parr\ \mathit{И}y.\ \overline{xz}\ \big) \triangleleft ([\![q]\!]_\pi \multimap \mathit{И}y.[\![p]\!]_\pi)}\ \text{extrude new}}{\big(\ xz \triangleleft \overline{[\![q]\!]_\pi}\big) \parr \big(\mathit{И}y.\ \overline{xz} \triangleleft \mathit{И}y.[\![p]\!]_\pi\big)}\ \text{sequence}}{\big(\ xz \triangleleft \overline{[\![q]\!]_\pi}\big) \parr \mathit{И}y.\big(\ \overline{xz} \triangleleft [\![p]\!]_\pi\big)}\ \text{medial new}$$

In the above proof, after removing the highlighted atoms xz and $\overline{xz}$, instances of the rules *sequence*, *medial new*, *extrude new*, and *tidy new* are deleted, yielding a proof of $[\![q]\!]_\pi \multimap \mathit{И}y.[\![p]\!]_\pi$.

5.1.2. *Scenario for inducing an internal interaction involving free output.* In the second scenario, the atom xz involved in the bottommost *atomic interaction* in a left proof can be traced to a free output and input action prefix in a process that can perform a τ-transition of the form $\mathcal{P}^0\{\ \mathcal{P}^1\{\ \overline{x}z.p\ \} \mid \mathcal{P}^2\{\ x(y).q\ \}\ \}$. This scenario allows us to induce τ-transitions in the process on the right hand side of a linear implication, which is essential for *weak* simulations.

Lemma 5.4. Assume $\vdash\ \mathcal{L}\{\ \circ\ \}$ holds and $\dfrac{\mathcal{L}\{\ \overline{xz} \parr xz\ \}}{\mathcal{K}\{\ [\![s]\!]_\pi \multimap \big[\!\big[\mathcal{P}^0\{\ \mathcal{P}^1\{\ \overline{x}z.p\ \} \mid \mathcal{P}^2\{\ x(y).q\ \}\ \}\big]\!\big]_\pi\ \}}$ is a derivation in the *cooling fragment* where s, p and q are π-calculus processes, $\mathcal{P}^i\{\ \cdot\ \}$ are process contexts and $\mathcal{L}\{\ \cdot\ \}$ is a left context, where atoms xz in the premiss of the derivation can be traced to the head of the sub-processes indicated in the conclusion, and assume x and z are not bound by $\mathcal{P}^1\{\ \cdot\ \}$ and $\mathcal{P}^2\{\ \cdot\ \}$.

Under the above assumptions, we have: $\vdash \mathcal{K}\{\ [\![s]\!]_\pi \multimap \big[\!\big[\mathcal{P}^0\{\ \mathcal{P}^1\{\ p\ \} \mid \mathcal{P}^2\{\ q\{^z/_y\}\ \}\ \}\big]\!\big]_\pi\ \}$.

To establish the above lemma, we require the decomposition result for existential quantifiers in Lemma 4.14, in order to handle the existential quantifier in the embedding of the input. Also, observe that, if formula $\exists x.P$ is in a left context, it will never be the case that $\exists x.P$ is nested inside another existential quantifier; where a nested quantifier would be able to influence the term introduced.

By the above observations, the derivation in the cooling fragment must be of the following form, for some left context $\mathcal{L}'\{\ \cdot\ \}$ and formula R where the existential quantifier from the input action is underlined and each $\mathcal{L}^i\{\ \cdot\ \}$ is structurally congruent to an embedding of the respective $\mathcal{P}^i\{\ \cdot\ \}$. Thereby, the conclusion of the following derivation is structurally congruent to $\mathcal{K}\{\ [\![s]\!]_\pi \multimap \big[\!\big[\mathcal{P}^0\{\ \mathcal{P}^1\{\ \overline{x}z.p\ \} \mid \mathcal{P}^2\{\ x(y).q\ \}\ \}\big]\!\big]_\pi\ \}$.

$$\dfrac{\dfrac{\dfrac{\mathcal{L}\{\ \overline{xz} \parr xz\ \}}{\mathcal{L}'\{\ R\{^z/_y\}\ \}}\ \text{using the } \textit{cooling} \text{ fragment only}}{\mathcal{L}'\{\ \exists y.\ R\ \}}\ \text{by the } \textit{select1} \text{ rule}}{\mathcal{K}\{\ [\![s]\!]_\pi \multimap \mathcal{L}^0\{\ \mathcal{L}^1\{\ \overline{xz} \triangleleft [\![p]\!]_\pi\ \} \parr \mathcal{L}^2\{\ \exists y.\ (xy \triangleleft [\![q]\!]_\pi)\ \}\ \}\ \}}\ \text{using the } \textit{cooling} \text{ fragment only}$$

Since $\mathcal{P}^1\{\ \cdot\ \}$ and $\mathcal{P}^2\{\ \cdot\ \}$ do not bind x and z; we have that $\mathcal{L}^1\{\ \cdot\ \}$ and $\mathcal{L}^2\{\ \cdot\ \}$ do not bind x

and z. Thereby, we can appeal to Lemma 4.14 to construct the following derivation in the cooling fragment of BV1.

$$\dfrac{\dfrac{\mathcal{L}\{\ \overline{xz} \parr xz\ \}}{\mathcal{L}'\{\ R\{^{z}/_{x}\}\ \}}\ \text{a derivation in the } \textit{cooling} \text{ fragment}}{\mathcal{K}\Big\{\ [\![s]\!]_\pi \multimap \mathcal{L}^0\Big\{\ \mathcal{L}^1\{\ \overline{xz} \triangleleft [\![p]\!]_\pi\ \} \parr \mathcal{L}^2\Big\{\ xz \triangleleft \Big[\!\Big[q\big\{^{z}/_{y}\big\}\Big]\!\Big]_\pi\ \Big\}\ \Big\}\ \Big\}}\ \text{a derivation in the } \textit{cooling} \text{ fragment}$$

We can now, similarly to Lemma 5.3, delete the instances of the atoms $\overline{xz}$ and xz involved in the bottommost interaction everywhere in the proof, while removing redundant rules. This leads to a left proof of the following formula, as desired: $\mathcal{K}\Big\{\ [\![s]\!]_\pi \multimap \Big[\!\Big[\mathcal{P}^0\Big\{\ \mathcal{P}^1\{\ p\ \} \mid \mathcal{P}^2\Big\{\ q\big\{^{z}/_{y}\big\}\ \Big\}\ \Big\}\Big]\!\Big]_\pi\ \Big\}$.

5.1.3. *Scenario for inducing internal interaction involving bound output.* Consider when both atoms of the bottommost *atomic interaction* rule appear in the embedding of a process of the form $\mathcal{P}^0\Big\{\ \mathcal{P}^1\Big\{\ \nu z.\mathcal{P}^2\{\ \overline{x}z.p\ \}\ \Big\} \mid \mathcal{P}^3\{\ x(z).q\ \}\ \Big\}$, where z is bound by *new*. In this scenario, clearly a τ-transition can be induced as formally stated in the following lemma.

Lemma 5.5. Assume $\vdash \mathcal{L}\{\circ\}$ holds and $\dfrac{\mathcal{L}\{\ \overline{xz} \parr xz\ \}}{\mathcal{K}\Big\{\ [\![s]\!]_\pi \multimap \Big[\!\Big[\mathcal{P}^0\Big\{\ \mathcal{P}^1\Big\{\ \nu z.\mathcal{P}^2\{\ \overline{x}z.p\ \}\ \Big\} \mid \mathcal{P}^3\{\ x(z).q\ \}\ \Big\}\Big]\!\Big]_\pi\ \Big\}}$ is a derivation in the *cooling* fragment, where x and z are not bound by $\mathcal{P}^1\{\ \cdot\ \}$, $\mathcal{P}^2\{\ \cdot\ \}$, or $\mathcal{P}^3\{\ \cdot\ \}$ and z is fresh for $\mathcal{P}^1\{\ \cdot\ \}$ and $\mathcal{P}^3\{\ \cdot\ \}$.

In this scenario, $\vdash \mathcal{K}\Big\{\ [\![s]\!]_\pi \multimap \Big[\!\Big[\mathcal{P}^0\Big\{\ \nu z.\Big(\mathcal{P}^1\Big\{\ \mathcal{P}^2\{\ p\ \}\ \Big\} \mid \mathcal{P}^3\{\ q\ \}\Big)\ \Big\}\Big]\!\Big]_\pi\ \Big\}$ holds.

To establish the above lemma, firstly we expand the scope of nominal quantifiers, to obtain a proof of a form similar to in the previous scenario. By Lemma 4.13, we can expand the scope of quantifiers, to obtain the following derivation in the cooling fragment.

$$\dfrac{\mathcal{L}\{\ \overline{xz} \parr xz\ \}}{\mathcal{K}\Big\{\ [\![s]\!]_\pi \multimap \Big[\!\Big[\mathcal{P}^0\Big\{\ \nu z.\Big(\mathcal{P}^1\Big\{\ \mathcal{P}^2\{\ \overline{x}z.p\ \}\ \Big\} \mid \mathcal{P}^3\{\ x(z).q\ \}\Big)\ \Big\}\Big]\!\Big]_\pi\ \Big\}}\ \text{a derivation in the } \textit{cooling} \text{ fragment}$$

Hence, by following the same strategy as for Lemma 5.4, we have a procedure for constructing a proof of the following formula, as required: $\mathcal{K}\Big\{\ [\![s]\!]_\pi \multimap \Big[\!\Big[\mathcal{P}^0\Big\{\ \nu z.\Big(\mathcal{P}^1\Big\{\ \mathcal{P}^2\{\ p\ \}\ \Big\} \mid \mathcal{P}^3\{\ q\ \}\Big)\ \Big\}\Big]\!\Big]_\pi\ \Big\}$.

5.1.4. *Scenario inducing a bound output transition.* Consider the scenario where the bottommost *atomic interaction* rule applies to an atom that can be traced to process embeddings of the form $\nu z.\overline{x}z.s$ and $\mathcal{P}^0\Big\{\ \nu z.\mathcal{P}^1\{\ \overline{x}z.q\ \}\ \Big\}$ — two processes ready to output a fresh name on channel x.

Lemma 5.6. Suppose, $\dfrac{\mathcal{L}\{\ xz \parr \overline{xz}\ \}}{\mathcal{K}\Big\{\ [\![\nu z.\overline{x}z.s]\!]_\pi \multimap \Big[\!\Big[\mathcal{P}^0\Big\{\ \nu z.\mathcal{P}^1\{\ \overline{x}z.q\ \}\ \Big\}\Big]\!\Big]_\pi\ \Big\}}$ is a derivation in the cooling fragment of BV1, where s and q are π-calculus process, $\mathcal{L}\{\ \cdot\ \}$ is a left context, $\mathcal{P}^0\{\ \cdot\ \}$ and $\mathcal{P}^1\{\ \cdot\ \}$ are process contexts that do not bind x and z; and z is fresh for $\mathcal{P}^0\{\ \cdot\ \}$. Also, assume $\mathcal{L}\{\circ\}$ has a left proof.

In this scenario, $\vdash \mathcal{K}\Big\{\ И z.\Big([\![s]\!]_\pi \multimap \Big[\!\Big[\mathcal{P}^0\Big\{\ \mathcal{P}^1\{\ q\ \}\ \Big\}\Big]\!\Big]_\pi\Big)\ \Big\}$ holds.

The proof is similar to the previous scenario where a bound output is involved in an interaction. By applying Lemma 4.13, we can pull the quantifier $И z$ to the outermost level, removing

corresponding *wen* operators to obtain a derivation in the cooling fragment of the following form.

$$\dfrac{\mathcal{L}\{\ xz \parr \overline{xz}\ \}}{\mathcal{K}\Big\{\ \text{И}z.\Big(\big(xz \triangleleft \overline{[\![s]\!]_\pi}\big) \parr \Big[\!\Big[\mathcal{P}^0\Big\{\ \mathcal{P}^1\{\ \overline{x}z.q\ \}\ \Big\}\Big]\!\Big]_\pi\Big)\ \Big\}}\ \text{a derivation in the } \textit{cooling} \text{ fragment}$$

Hence, by deleting the interacting atoms, a proof of $\mathcal{K}\Big\{\ \text{И}z.\Big([\![s]\!]_\pi \multimap \Big[\!\Big[\mathcal{P}^0\Big\{\ \mathcal{P}^1\{\ q\ \}\ \Big\}\Big]\!\Big]_\pi\Big)\ \Big\}$, can be constructed as required.

5.1.5. *Scenario inducing an input transition.* The scenario for inputs is where the bottommost *atomic interaction* in a left proof involves an atom that can be traced to the prefix in processes embeddings of the following form in the conclusion of a proof: $x(z).s$ and $\mathcal{P}\{\ x(z).q\ \}$, each ready to perform an input action.

Lemma 5.7. Assume $\vdash \mathcal{L}\{\ \circ\ \}$ holds and $\dfrac{\mathcal{L}\{\ \overline{xz} \parr xz\ \}}{\mathcal{K}\{\ [\![x(z).s]\!]_\pi \multimap [\![\mathcal{P}\{\ x(z).q\ \}]\!]_\pi\ \}}$ is a derivation in the cooling fragment, where x and z are not bound by process context $\mathcal{P}\{\ \cdot\ \}$ and z does not appear free in $\mathcal{P}\{\ x(z).q\ \}$ and $\mathcal{L}\{\ \cdot\ \}$ is a left context.

In this scenario $\vdash \mathcal{K}\{\ \forall z.([\![s]\!]_\pi \multimap [\![\mathcal{P}\{\ q\ \}]\!]_\pi)\ \}$ holds.

Similarly to the scenarios involving bound output actions, the trick to establish the above lemma is to first move the universal quantifier binding the name in the input action out of the way. Since $\vdash \forall z.\Big(\overline{xz} \triangleleft \overline{[\![s]\!]_\pi}\Big) \parr [\![\mathcal{P}\{\ x(z).q\ \}]\!]_\pi \multimap \forall z.\Big(\Big(\overline{xz} \triangleleft \overline{[\![s]\!]_\pi}\Big) \parr [\![\mathcal{P}\{\ x(z).q\ \}]\!]_\pi\Big)$ holds, by Theorem 3.2, we can construct a derivation in the cooling fragment of the following form.

$$\dfrac{\mathcal{L}\{\ \overline{xz} \parr xz\ \}}{\mathcal{K}\Big\{\ \forall z.\Big(\Big(\overline{xz} \triangleleft \overline{[\![s]\!]_\pi}\Big) \parr [\![\mathcal{P}\{\ x(z).q\ \}]\!]_\pi\Big)\ \Big\}}\ \text{a derivation in the } \textit{cooling} \text{ fragment}$$

Observe that the above derivation in the cooling fragment must be of the following form, where $\mathcal{L}\{\ \cdot\ \}$ is structurally equivalent to the embedding of $\mathcal{P}\{\ \cdot\ \}$.

$$\dfrac{\dfrac{\dfrac{\mathcal{L}\{\ \overline{xz} \parr xz\ \}}{\mathcal{L}'\{\ R\ \}}\ \text{a derivation in the } \textit{cooling} \text{ fragment}}{\mathcal{L}'\Big\{\ \exists z.\ R\ \Big\}}\ \text{by the } \textit{select1} \text{ rule}}{\mathcal{K}\Big\{\ \forall z.\Big(\Big(\overline{xz} \triangleleft \overline{[\![s]\!]_\pi}\Big) \parr \mathcal{L}\Big\{\ \exists z.\ (xz \triangleleft [\![Q]\!]_\pi)\ \Big\}\Big)\ \Big\}}\ \text{a derivation in the } \textit{cooling} \text{ fragment}$$

Hence by Lemma 4.14 there exists a derivation in the cooling fragment of the following form.

$$\dfrac{\dfrac{\mathcal{L}\{\ \overline{xz} \parr xz\ \}}{\mathcal{L}'\{\ R\ \}}\ \text{a derivation in the } \textit{cooling} \text{ fragment}}{\mathcal{K}\Big\{\ \forall z.\Big(\Big(\overline{xz} \triangleleft \overline{[\![s]\!]_\pi}\Big) \parr \mathcal{L}\{\ xz \triangleleft [\![q]\!]_\pi\ \}\Big)\ \Big\}}\ \text{a derivation in the } \textit{cooling} \text{ fragment}$$

By removing the pair of atoms $\overline{xz}$ and xz from the above derivation, we can construct the derivation $\dfrac{\mathcal{L}\{\ \circ\ \}}{\mathcal{K}\Big\{\ \forall z.\Big(\overline{[\![s]\!]_\pi} \parr \mathcal{L}\{\ [\![q]\!]_\pi\ \}\Big)\ \Big\}}$. Thereby, since we assumed $\mathcal{L}\{\ \circ\ \}$ has a proof, we can construct a proof of $\mathcal{K}\{\ \forall z.([\![s]\!]_\pi \multimap [\![\mathcal{P}\{\ q\ \}]\!]_\pi)\ \}$.

5.2. *Proof of the soundness of linear implication with respect to weak simulation.*

We are now ready to prove the main result of this paper. We require the following intermediate definition mapping histories to a list of quantifiers $И$ and $\forall$.

Definition 5.8. Let the abbreviation $\mathcal{H}h.P$ be the formula defined such that $\mathcal{H}\epsilon.P = P$ and, inductively, $\mathcal{H}(h \cdot x^o).P = \mathcal{H}h.Иx.P$ and $\mathcal{H}\left(h \cdot x^i\right).P = \mathcal{H}h.\forall x.P$.

Given a history h, we define a trimming function $\lceil h \rceil$ as follows:

— $\lceil \epsilon \rceil = \epsilon$,
— If $h = h'.x^\bullet$, where $\bullet \in \{i, o\}$, and $x \notin n(h')$ then $\lceil h \rceil = \lceil h' \rceil.x^\bullet$. Otherwise, $\lceil h \rceil = \lceil h' \rceil$.

The embedding of histories in Definition 5.8 satisfies the following lemma.

Lemma 5.9. If σ respects h and $h' = \lceil h \rceil$, then if $\vdash \mathcal{H}h.P$ holds then $\vdash \mathcal{H}h'.P\sigma$.

The following proposition combines the intermediate results established throughout this paper: cut elimination (Theorem 3.2); the mapping of labelled transitions to proofs (Lemma 3.11); the normalisation results Proposition 4.11, Lemmas 4.13 and 4.14; the above Lemma 5.9; as well as the observations made in Lemmas 5.3 to 5.7 in the previous section.

Proposition 5.10. Define ternary relation $\mathcal{R}$ such that $p \; \mathcal{R}^h \; q$ whenever $\vdash \mathcal{H}h.(\llbracket p \rrbracket_\pi \multimap \llbracket q \rrbracket_\pi)$. The relation $\mathcal{R}$ is a complete weak open simulation.

Proof. Assume that $\vdash \mathcal{H}h.(\llbracket p \rrbracket_\pi \multimap \llbracket q \rrbracket_\pi)$ holds, i.e. $p \; \mathcal{R}^h \; q$. There are six cases to consider to show $\mathcal{R}$ is closed under the definition of complete weak open simulation.

To show preservation under respectful substitutions, assume σ respects h, by Lemma 5.9, $\vdash \mathcal{H}\lceil h\sigma \rceil.(\llbracket p \rrbracket_\pi \, \sigma \multimap \llbracket q \rrbracket_\pi \, \sigma)$; hence, by Lemma 3.10, $\vdash \mathcal{H}\lceil h\sigma \rceil.(\llbracket p\sigma \rrbracket_\pi \multimap \llbracket q\sigma \rrbracket_\pi)$, as required.

To show preservation under τ-transitions, assume $p \xrightarrow{\tau} p'$, and observe $\vdash \llbracket p' \rrbracket_{\pi I} \multimap \llbracket p \rrbracket_{\pi I}$, by Lemma 3.11. Therefore, by Theorem 3.2, $\vdash Иx_1, \ldots Иx_n.(\llbracket p' \rrbracket_\pi \multimap \llbracket q \rrbracket_\pi)$ holds, as required.

Check termination potential is preserved. Assume, in this case, $p\checkmark$ holds. Now, consider, more generally, r such that, $\mathcal{H}h.(\llbracket p \rrbracket_\pi \multimap \llbracket r \rrbracket_\pi)$. By Proposition 4.11, $\mathcal{H}h.(\llbracket p \rrbracket_\pi \multimap \llbracket r \rrbracket_\pi)$ has a left proof, and, whenever there is at least one atom, by Lemma 4.12, the bottommost rule of the interaction fragment can arranged to be an instance of the *atomic interaction* rule. Since there are no atoms in p, three possible scenarios can occur:

—There are no atoms in $\llbracket r \rrbracket_\pi$ and hence process r is also such that $r\checkmark$.
—Process q is of the form $\mathcal{P}^1\left\{ \mathcal{P}^2\{ \; \overline{x}y.r_1 \; \} \mid \mathcal{P}^3\{ \; x(z).r_2 \; \} \right\}$, where x, y and z are not bound by $\mathcal{P}^2\{ \; \cdot \; \}$ and $\mathcal{P}^3\{ \; \cdot \; \}$; and, furthermore, the atoms in the bottommost *atomic interaction* can be traced to the atoms in the output and input prefixes identified. In this case, by Lemma 5.4, we can construct a proof of $\vdash \mathcal{H}h.\left(\llbracket p \rrbracket_\pi \multimap \left\llbracket \mathcal{P}^1\left\{ \mathcal{P}^2\{ \; r_1 \; \} \mid \mathcal{P}^3\{ \; r_2\{^y/_z\} \; \} \right\} \right\rrbracket_\pi\right)$. Furthermore, by Lemma 5.2, $\mathcal{P}^1\left\{ \mathcal{P}^2\{ \; \overline{x}y.r_1 \; \} \mid \mathcal{P}^3\{ \; x(z).r_2 \; \} \right\} \xrightarrow{\tau} \mathcal{P}^1\left\{ \mathcal{P}^2\{ \; r_1 \; \} \mid \mathcal{P}^3\{ \; r_2\{^y/_z\} \; \} \right\}$.
—Process q is of the form $q = \mathcal{P}^0\left\{ \mathcal{P}^1\left\{ \nu z.\mathcal{P}^2\{ \; \overline{x}z.r_1 \; \} \right\} \mid \mathcal{P}^3\{ \; x(z).r_2 \; \} \right\}$, where x and z are not bound by $\mathcal{P}^1\{ \; \cdot \; \}$, $\mathcal{P}^2\{ \; \cdot \; \}$ and $\mathcal{P}^3\{ \; \cdot \; \}$, z is fresh for $\mathcal{P}^1\{ \; \cdot \; \}$ and $\mathcal{P}^3\{ \; \cdot \; \}$; and, furthermore, the atoms in the bottommost *atomic interaction* can be traced to the atoms in the output and input prefixes identified. In this case, by Lemma 5.5, we can construct a proof of the following: $\mathcal{H}h.\left(\llbracket p \rrbracket_\pi \multimap \left\llbracket \mathcal{P}^0\left\{ \nu z.\left(\mathcal{P}^1\left\{ \mathcal{P}^2\{ \; r_1 \; \} \right\} \mid \mathcal{P}^3\{ \; r_2 \; \}\right) \right\} \right\rrbracket_\pi\right)$. So, by Lemma 5.2, we have the following: $\mathcal{P}^0\left\{ \mathcal{P}^1\left\{ \nu z.\mathcal{P}^2\{ \; \overline{x}z.r_1 \; \} \right\} \mid \mathcal{P}^3\{ \; x(z).r_2 \; \} \right\} \xrightarrow{\tau} \mathcal{P}^0\left\{ \nu z.\left(\mathcal{P}^1\left\{ \mathcal{P}^2\{ \; r_1 \; \} \right\} \mid \mathcal{P}^3\{ \; r_2 \; \}\right) \right\}$.

Since τ-transitions strictly decrease the size of the processes and $\mathcal{H}h.(\llbracket p \rrbracket_\pi \multimap \llbracket q \rrbracket_\pi)$ only finitely many τ-transitions $q = q_1 \xrightarrow{\tau} q_2 \ldots \xrightarrow{\tau} q_n$, can be applied. Furthermore, $\mathcal{H}h.(\llbracket p \rrbracket_\pi \multimap \llbracket q_n \rrbracket_\pi)$ hence, since no further τ-transition can be enabled, by the above case analysis, $q_n{\checkmark}$ holds. Hence we can construct q_n such that $q \Longrightarrow q_n$ and $q_n{\checkmark}$ holds, as required.

Check preservation under free outputs. Assume, in this case, $p \xrightarrow{\overline{x}z} p'$. Firstly consider, more generally, judgements of the form $\vdash \mathcal{H}h.(\llbracket \overline{x}z.p' \rrbracket_\pi \multimap \llbracket r \rrbracket_\pi)$. By Proposition 4.11, we can construct a left proof such that, by Lemma 4.12, the bottommost rule in the interaction fragment is an *atomic interaction* rule. There are two possibilities:

—As described in detail in the previous case, atoms in the bottommost *atomic interaction* rule correspond to an input and output prefix in r; hence, by Lemma 5.4 or Lemma 5.5 and also appealing to Lemma 5.2, we can construct r' such that $r \xrightarrow{\tau} r'$ and $\vdash \mathcal{H}h.(\llbracket \overline{x}z.p' \rrbracket_\pi \multimap \llbracket r' \rrbracket_\pi)$.

—Following Lemma 5.3, the atoms in the bottommost *atomic interaction* rule correspond to xz and an output prefix in r, where $r = \mathcal{P}\{\ \overline{x}z.s\ \}$ such that x and z are not bound by $\mathcal{P}\{\ \cdot\ \}$; and we can construct a proof of $\vdash \mathcal{H}h.(\llbracket p' \rrbracket_\pi \multimap \llbracket \mathcal{P}\{\ s\ \} \rrbracket_\pi)$. Also, by Lemma 5.2, $r \xrightarrow{\overline{x}z} \mathcal{P}\{\ s\ \}$.

By Lemma 3.11, $\vdash \llbracket \overline{x}z.p' \rrbracket_\pi \multimap \llbracket p \rrbracket_\pi$; hence, by Theorem 3.2, $\vdash \mathcal{H}h.(\llbracket \overline{x}z.p' \rrbracket_\pi \multimap \llbracket q \rrbracket_\pi)$. Since τ-transitions strictly decrease the size of the process, and $\vdash \mathcal{H}h.(\llbracket \overline{x}z.p' \rrbracket_\pi \multimap \llbracket q \rrbracket_\pi)$, finitely many τ-transitions can be applied $q = q_1 \xrightarrow{\tau} q_2 \xrightarrow{\tau} \ldots \xrightarrow{\tau} q_n$. Since no further τ-transition is possible, output transition $p \xrightarrow{\overline{x}z} q_n$ must be enabled and furthermore we have $\vdash \mathcal{H}h.\Big(\overline{\llbracket p' \rrbracket_\pi} \parr \llbracket q_n \rrbracket_\pi\Big)$. Thereby, we can construct q_n such that $q \xRightarrow{\overline{x}z} q_n$ and $p'\ \mathcal{R}^h\ q_n$ as required.

Check preservation under bound outputs. Consider the cases where $p \xrightarrow{\overline{x}[z]} p'$ such that z is fresh for p, q and h. Consider judgements of the form $\vdash \mathcal{H}h.(\llbracket \nu z.\overline{x}z.p' \rrbracket_\pi \multimap \llbracket r \rrbracket_\pi)$, by Proposition 4.11, we can construct a left proof such that the bottommost rule in the interaction fragment is an *atomic interaction* rule. There are two possibilities:

—As previously, Lemma 5.4 or Lemma 5.5 applies; thereby, appealing also to Lemma 5.2 there exists r' such that $r \xrightarrow{\tau} r'$ and $\vdash \mathcal{H}h.(\llbracket \nu z.\overline{x}z.p' \rrbracket_\pi \multimap \llbracket r' \rrbracket_\pi)$.

—The atoms in the bottommost *atomic interaction* rule correspond to the atom xz and an output in r, where $r = \mathcal{P}^1\Big\{\ \nu z.\mathcal{P}^2\{\ \overline{x}z.s\ \}\ \Big\}$ such that x and z are not bound by $\mathcal{P}^1\{\ \cdot\ \}$ or $\mathcal{P}^2\{\ \cdot\ \}$ and z is fresh for $\mathcal{P}^1\{\ \cdot\ \}$. In this case, following Lemma 5.6, we can construct a proof of $\vdash \mathcal{H}h.\Big(И z.\Big(\llbracket p' \rrbracket_\pi \multimap \Big\llbracket \mathcal{P}^1\Big\{\ \mathcal{P}^2\{\ s\ \}\ \Big\}\Big\rrbracket_\pi\Big)\Big)$ and furthermore, by Lemma 5.2, $r \xrightarrow{\overline{x}z} \mathcal{P}^1\Big\{\ \mathcal{P}^2\{\ s\ \}\ \Big\}$.

By Lemma 3.11, $\vdash \llbracket \nu z.\overline{x}z.p' \rrbracket_\pi \multimap \llbracket p \rrbracket_\pi$; hence, by Theorem 3.2, $\vdash \mathcal{H}h.(\llbracket \nu z.\overline{x}z.p' \rrbracket_\pi \multimap \llbracket q \rrbracket_\pi)$. Now, since τ-transitions strictly decrease the size of the process r only finitely many τ-transitions can be induced; after which, for some q' and r the output transition $r \xrightarrow{\overline{x}z} q'$ must be enabled such that $\vdash \mathcal{H}h.(И z.(\llbracket p' \rrbracket_\pi \multimap \llbracket q' \rrbracket_\pi))$; hence, by definition of the embedding of histories, $\vdash \mathcal{H}(h \cdot z^o).(\llbracket p' \rrbracket_\pi \multimap \llbracket q' \rrbracket_\pi)$. Thereby we have q' such that $q \xRightarrow{\overline{x}z} q'$ and $p'\ \mathcal{R}^{h \cdot z_o}\ q'$, as required.

Check preservation under inputs. Assume in this case $p \xrightarrow{x(z)} p'$ such that z is fresh for p, q and h. For a judgement of the form $\vdash \mathcal{H}h.(\llbracket x(z).p' \rrbracket_\pi \multimap \llbracket r \rrbracket_\pi)$, by Proposition 4.11, we can construct a left proof such that, Lemma 4.12, the bottommost rule in the interaction fragment is an *atomic interaction* rule; thereby identifying the following two possibilities:

—As in other cases, Lemma 5.4 or Lemma 5.5 applies, along with Lemma 5.2 so there exists r' such that $r \xrightarrow{\tau} r'$ and $\vdash \mathcal{H}h.(\llbracket x(z).p' \rrbracket_\pi \multimap \llbracket r' \rrbracket_\pi)$.

—The atoms in the bottommost *atomic interaction* rule correspond to the atom xz and an input in r such that $r = \mathcal{P}\{\ x(z).s\ \}$, where x and z are not bound by $\mathcal{P}\{\ \cdot\ \}$ and z is fresh for $\mathcal{P}\{\ \cdot\ \}$.

In this case, Lemma 5.7 applies and hence $\vdash \mathcal{H}h.(\forall z.([\![p']\!]_\pi \multimap [\![\mathcal{P}\{\ s\ \}]\!]_\pi))$. Furthermore, by Lemma 5.2, $r \xrightarrow{x(z)} \mathcal{P}\{\ s\ \}$.

By Lemma 3.11, $\vdash [\![x(z).p']\!]_\pi \multimap [\![p]\!]_\pi$; hence, by Theorem 3.2, $\vdash \mathcal{H}h.([\![x(z).p']\!]_\pi \multimap [\![q]\!]_\pi)$. Since τ-transitions strictly decrease the size of a process, finitely many τ-transitions can be applied reaching s such that, for some q', $s \xrightarrow{x(z)} q'$ such that $\vdash \mathcal{H}h.(\forall z.([\![p']\!]_\pi \multimap [\![q']\!]_\pi))$, thereby by the definition of embeddings of histories $\vdash \mathcal{H}\left(h \cdot z^i\right).([\![p']\!]_\pi \multimap [\![q']\!]_\pi)$. Thereby we can construct q' such that $q \overset{x(z)}{\Longrightarrow} q'$ and $p' \mathrel{\mathcal{R}^{h\cdot z^i}} q'$, as required.

Thus $\mathcal{R}$ is closed under the definition of a complete weak open simulation. □

Theorem 3.8, follows immediately from the above proposition as follows.

Proof of Theorem 3.8. Observe, if $\vdash [\![p]\!]_\pi \multimap [\![q]\!]_\pi$ and $\mathrm{fv}(p) \cup \mathrm{fv}(q) \subseteq \{x_0, \ldots, x_n\}$, then $\vdash \forall x_0. \ldots \forall x_n. ([\![p]\!]_\pi \multimap [\![q]\!]_\pi)$ hence, by definition, $\vdash \mathcal{H}x_0^i \cdot \ldots \cdot x_n^i.([\![p]\!]_\pi \multimap [\![q]\!]_\pi)$. Therefore, by Proposition 5.10, $p \mathrel{\mathcal{R}^{x_0^i \cdot \ldots \cdot x_n^i}} q$ for a complete weak open simulation $\mathcal{R}$; hence $p \lesssim q$. □

5.3. *Example of constructing a weak simulation from the proof of a linear implication.*

All proofs in this paper are constructive, hence Theorem 3.8 provides a procedure for extracting a complete weak open simulation from a proof of an implication. We illustrate this proceedure on the following example of a left proof.

$$
\begin{array}{ll}
\circ & \\ \hline
\forall a.\forall b.\text{И}k.\circ & \text{tidy} \\ \hline
\forall a.\forall b.\text{И}k.\left(ak \parr \overline{ak}\right) & \text{atomic interaction} \\ \hline
\forall a.\forall b.\text{И}k.\left(\left(\overline{bk} \parr bk\right) \triangleleft \left(ak \parr \overline{ak}\right)\right) & \text{atomic interaction} \\ \hline
\forall a.\forall b.\text{И}k.\left(\left(\overline{bk} \triangleleft ak\right) \parr \left(bk \triangleleft \overline{ak}\right)\right) & \text{sequence} \\ \hline
\forall a.\forall b.\text{И}k.\left(\left(\overline{bk} \triangleleft ak\right) \parr \exists y.(by \triangleleft \overline{ay})\right) & \text{select1} \\ \hline
\forall a.\forall b.\text{И}k.\left(\overline{bk} \parr ak \parr \exists y.(by \triangleleft \overline{ay})\right) & \text{sequence} \\ \hline
\forall a.\forall b.\left(\left(ab \parr \overline{ab}\right) \triangleleft \text{И}k.\left(\overline{bk} \parr ak \parr \exists y.(by \triangleleft \overline{ay})\right)\right) & \text{atomic interaction} \\ \hline
\forall a.\forall b.\left(\left(ab \parr \overline{ab}\right) \triangleleft \left(\text{И}k.\left(\overline{bk} \parr ak\right) \parr \exists y.(by \triangleleft \overline{ay})\right)\right) & \text{extrude new} \\ \hline
\forall a.\forall b.\left(\left(ab \parr \overline{ab}\right) \triangleleft \left(\text{И}k.\overline{bk} \parr \text{Э}k.ak \parr \exists y.(by \triangleleft \overline{ay})\right)\right) & \text{close} \\ \hline
\forall a.\forall b.\left(\left(ab \triangleleft \text{И}k.\overline{bk}\right) \parr \left(\overline{ab} \triangleleft (\text{Э}k.ak \parr \exists y.(by \triangleleft \overline{ay}))\right)\right) & \text{sequence} \\ \hline
\forall a.\forall b.\left(\exists x.\left(ax \triangleleft \text{И}k.\overline{xk}\right) \parr \left(\overline{ab} \triangleleft (\text{Э}k.ak \parr \exists y.(by \triangleleft \overline{ay}))\right)\right) & \text{select1} \\ \hline
\forall a.\forall b.\left(\text{Э}k.ak \parr \exists x.\left(ax \triangleleft \text{И}k.\overline{xk}\right) \parr \exists y.(by \triangleleft \overline{ay}) \parr \overline{ab}\right) & \text{sequence} \\ \hline
\mathcal{H}a^i \cdot b^i.([\![\nu k.\overline{a}k]\!]_\pi \multimap [\![a(x).\nu k.\overline{x}k \mid b(y).\overline{a}y \mid \overline{a}b]\!]_\pi) & \text{definition}
\end{array}
$$

In the above proof, the bottommost interacting atoms and corresponding existential quantifier are highlighted by boxes. Following Scenario 5.4: firstly, we apply Lemma 4.14 to apply a sub-

stitution in place of the highlighted *select1* rule; and, secondly, we remove the highlighted atoms from the proof. Thereby we construct the following left proof.

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\circ}{\forall a.\forall b.\ И k.\ \circ}\ \text{tidy}}{\forall a.\forall b.\ И k.\ (ak \parr \overline{ak})}\ \text{atomic interaction}}{\forall a.\forall b.\ И k.\ ((\overline{bk} \parr bk) \triangleleft (ak \parr \overline{ak}))}\ \text{atomic interaction}}{\forall a.\forall b.\ И k.\ ((\overline{bk} \triangleleft ak) \parr (bk \triangleleft \overline{ak}))}\ \text{sequence}}{\forall a.\forall b.\ И k.\ ((\overline{bk} \triangleleft ak) \parr \exists y.\ (by \triangleleft \overline{ay}))}\ \text{select1}}{\forall a.\forall b.\ И k.\ (\overline{bk} \parr ak \parr \exists y.\ (by \triangleleft \overline{ay}))}\ \text{sequence}}{\forall a.\forall b.\ (И k.\ (\overline{bk} \parr ak) \parr \exists y.\ (bk \triangleleft \overline{ay}))}\ \text{extrude new}}{\forall a.\forall b.\ (\ni k.\ ak \parr И k.\ \overline{bk} \parr \exists y.\ (by \triangleleft \overline{ay}))}\ \text{close}}{\mathcal{H}a^i \cdot b^i.([\![\nu k.\overline{a}k]\!]_\pi \multimap [\![\nu k.\overline{b}k \mid b(y).\overline{a}y]\!]_\pi)}\ \text{definition}$$

Again, in the above proof, the bottommost interacting atoms and corresponding quantifers appear in boxes. Following Scenario 5.5: firstly, we apply Lemma 4.13 to expand the scope of the new quantifier in the process embedding on the right of the implication; secondly, we apply Lemma 4.14 to replace the highlighted existential quantifier with a substitution; and, thirdly, we remove the highlighted atoms from the proof. This results in the following left proof.

$$\dfrac{\dfrac{\dfrac{\dfrac{\circ}{\forall a.\forall b.\ И k.\ \circ}\ \text{tidy}}{\forall a.\forall b.\ И k.\ (ak \parr \overline{ak})}\ \text{atomic interaction}}{\forall a.\forall b.\ (\ni k.\ ak \parr И k.\ (\circ \parr \overline{ak}))}\ \text{close}}{\mathcal{H}a^i \cdot b^i.([\![\nu k.\overline{a}k]\!]_\pi \multimap [\![\nu k.\,(1 \mid \overline{a}k)]\!]_\pi)}\ \text{definition}$$

Following Scenario 5.6, using Lemma 4.13, the above proof can be transformed to a left proof of $\mathcal{H}a^i \cdot b^i \cdot k^o.([\![1]\!]_\pi \multimap [\![1 \mid 1]\!]_\pi)$, employing tidy rules only. Notice the scope of the nominal quantifiers in both process embeddings have been enlarged such that they become part of the history. Thereby on the left of the implication in the conclusion of each proof we have constructed transition $\nu k.\overline{a}k \xrightarrow{\overline{a}[k]} 1$, and on the right the following series of transitions.

$$a(x).\nu k.\overline{x}k \mid b(y).\overline{a}y \mid \overline{a}b \xrightarrow{\tau} \nu k.\overline{b}k \mid b(y).\overline{a}y \xrightarrow{\tau} \nu k.\,(1 \mid \overline{a}k) \xrightarrow{\overline{a}[k]} 1 \mid 1$$

Notice $(1 \mid 1)\checkmark$, so termination potential is preserved. Hence relation $\mathcal{S}$, consisting of triples $\nu k.\overline{a}k\ \mathcal{S}^{a^i \cdot b^i}\ a(x).\nu k.\overline{x}k \mid b(y).\overline{a}y \mid \overline{a}b$ and $1\ \mathcal{S}^{a^i \cdot b^i \cdot k^o}\ 1 \mid 1$, is a complete weak open simulation.

6. The Versatility of the Processes-as-Formulae Approach

We highlight the versatility of our approach by showing variants of the π-calculus can also be embedded as processes in BV1 such that implication defines a sound preorder over processes. We also discuss a surprising strict inequality in BV1 and how it arises naturally when both unrestricted input and private input coexist in a process calculus.

6.1. *Extending results to the internal π-calculus*

We briefly outline how the techniques in this paper can be extended to the internal π-calculus Sangiorgi (1996a), called the πI-calculus, where inputs are guaranteed to be private. The syntax and semantics for a fragment of the πI-calculus is presented in Fig. 6. The private restriction on input names is enforced by using $\Im$ in place of $\exists$ in the embedding of private inputs. We are able to obtain that, for our embedding of πI-calculus processes as formulae in BV1, linear implication is sound with respect to complete weak simulation, and hence also completed trace inclusion.

The embedding of πI-calculus processes as BV1 formulae is defined as follows.

$$[\![1]\!]_{\pi I} = \circ \qquad [\![p \mid q]\!]_{\pi I} = [\![p]\!]_{\pi I} \parr [\![q]\!]_{\pi I} \qquad [\![\nu x.p]\!]_{\pi I} = И x.[\![p]\!]_{\pi I}$$
$$[\![x[z].p]\!]_{\pi I} = \Im z.(xz \triangleleft [\![p]\!]_{\pi I}) \qquad [\![\overline{x}[z].p]\!]_{\pi I} = И z.(\overline{xz} \triangleleft [\![p]\!]_{\pi I})$$

Recall from the discussion in Section 3.2 that we use the syntax $x[z]$ to represent *private input* in the πI-calculus to syntactically disambiguate from the semantically distinct input $x(z)$ in the π-calculus. The output process $\overline{x}[z].p$ behaves much like the output $\nu z.\overline{x}z.p$ in the π-calculus, where the private name binder νz appears immediately before the action that outputs the name z.

$$\begin{array}{lll} p ::= & 1 & \text{(success)} \\ & \nu x.p & \text{(nu)} \\ & \overline{x}[z].p & \text{(private input)} \\ & x[z].p & \text{(private output)} \\ & p \mid p & \text{(par)} \\ \pi ::= \tau \mid \overline{x}[z] \mid x[z] & & \text{(actions)} \end{array}$$

$$\frac{}{\overline{x}[z].p \xrightarrow{\overline{x}[z]} p} \qquad \frac{}{x[z].p \xrightarrow{x[z]} p} \qquad \frac{p \xrightarrow{\pi} q}{\nu x.p \xrightarrow{\pi} \nu x.q}\ x \notin \mathrm{n}(\pi)$$

$$\frac{p \xrightarrow{\overline{x}[z]} p' \quad q \xrightarrow{x[z]} q'}{p \mid q \xrightarrow{\tau} \nu z.(p' \mid q')} \qquad \frac{p \xrightarrow{\pi} r}{p \mid q \xrightarrow{\pi} r \mid q}\ \text{if } \pi = \overline{x}[z] \text{ or } \pi = x[z],\ z \# q$$

Fig. 6. Syntax and labelled transitions for the πI-calculus (plus symmetric rules for $p \mid q$). Function $n(.)$ is such that $n(x[z]) = n(\overline{x}[z]) = \{x, z\}$ and $n(\tau) = \emptyset$.

For every labelled transition there is a corresponding implication. As with the π-calculus, Lemma 3.11, the proof follows from cut elimination for BV1 (Theorem 3.2), by induction over the structure of the transition system.

Proposition 6.1. The following hold for πI-calculus processes.

— If $p \xrightarrow{\overline{x}[z]} q$, then $\vdash [\![\overline{x}[z].q]\!]_{\pi I} \multimap [\![p]\!]_{\pi I}$ holds.
— If $p \xrightarrow{x[z]} q$, then $\vdash [\![x[z].q]\!]_{\pi I} \multimap [\![p]\!]_{\pi I}$ holds.
— If $p \xrightarrow{\tau} q$ then $\vdash [\![q]\!]_{\pi I} \multimap [\![p]\!]_{\pi I}$.

We can define a fine notion of weak simulation for the πI-calculus (note the 'open' constraint is irrelevant for the πI-calculus, since no distinct free variable can be unified in any context).

Definition 6.2. A complete weak simulation is a relation, such that, if $p \mathcal{R} q$, the following hold:

— If $p\checkmark$ then there exists q' such that $q \Longrightarrow q'$ and $q'\checkmark$.
— If $p \xrightarrow{\tau} p'$ then there exists q' such that $q \xRightarrow{\tau} q'$ and $p' \mathcal{R} q'$.
— If $p \xrightarrow{\overline{x}[z]} p'$, where z fresh for p and q, there exists q' such that $q \xRightarrow{\overline{x}[z]} q'$ and $p' \mathcal{R} q'$.
— If $p \xrightarrow{x[z]} p'$, where z fresh for p and q, there exists q' such that $q \xRightarrow{x[z]} q'$ and $p' \mathcal{R} q'$.

Whenever there exists a complete weak simulation $\mathcal{R}$ such that $p \mathcal{R} q$, we write $p \preceq q$ and say p is simulated by q.

As for the π-calculus, we can established the soundness of linear implication with respect to complete weak simulation for the πI-calculus.

Theorem 6.3. For πI-calculus processes, If $\vdash [\![p]\!]_{\pi I} \multimap [\![q]\!]_{\pi I}$ then, $p \preceq q$.

The proof follows a similar strategy to Proposition 5.10. The proof appeals to Proposition 6.1, Theorem 3.2, Propositions 4.11 and Lemma 4.13, as well as a similar argument to Scenarios 5.5 and 5.6, involving actions with bound names.

6.2. *Explaining curiosities regarding name extrusion*

A curiosity of BV1 is that in general the scope of names can only by extruded and not contracted. By this we mean that, when $x \# Q$, there is **not** a proof of $И x.P \parr Q \multimap И x.(P \parr Q)$ in general. A logical reason for this restriction is that attempts to include an equivalence equating the formulae $И x.P \parr Q$ and $И x.(P \parr Q)$, where $x \# Q$, presented difficulties when proving cut elimination. Instead we only have the rule *extrude new* in Fig. 4. Most likely, the unidirectional nature of *extrude new* is for deeper reasons. To see this, observe that, with *extrude new* as an equality, instead of a unidirectional rule, the implication $[\![\nu z.\overline{a}z.\nu z.\overline{a}z]\!]_\pi \multimap [\![\nu z.\overline{a}z.\overline{a}z]\!]_\pi$ would **wrongly** be provable. Such an implication would be **unsound** with respect to weak simulation. The former process outputs two distinct names, but the later cannot.

The curiosity regarding scope extrusion, mentioned above, can also be explained in terms of the ability of BV1 to express both the π-calculus and the πI-calculus. To see this, suppose that we provide semantics that combining the π-calculus and the πI-calculus. Such a semantics should give a meaning to process $a[x].a[y] \mid \nu z.\overline{a}z.\overline{a}z$, where $a[x].a[y]$ here represents private inputs of the πI-calculus and $\nu z.\overline{a}z.\overline{a}z$ is a π-calculus term that outputs the same fresh name twice. Naïvely, we may **wrongly** think that the following two transitions form a completed trace.

$$\dfrac{\dfrac{}{a[x].a[y] \xrightarrow{a[x]} a[y]} \quad \dfrac{\dfrac{}{\overline{a}z.\overline{a}z \xrightarrow{\overline{a}z} \overline{a}z}}{\nu z.\overline{a}z.\overline{a}z \xrightarrow{\overline{a}[z]} \overline{a}z}}{a[x].a[y] \mid \nu z.\overline{a}z.\overline{a}z \xrightarrow{\tau} \nu z.(a[y] \mid \overline{a}z)} \qquad \dfrac{\dfrac{}{a[y] \xrightarrow{a[z]} 1} \quad \dfrac{\dfrac{}{\overline{a}z \xrightarrow{\overline{a}z} 1}}{\nu z.\overline{a}z \xrightarrow{\overline{a}[z]} 1}}{a[y] \mid \nu z.\overline{a}z \xrightarrow{\tau} \nu z.(1 \mid 1)}$$

Suppose, in this hybrid $\pi/\pi I$-calculus setting, we **wrongly** assume $\nu z.(a[y] \mid \overline{a}z)$ and $a[y] \mid \nu z.\overline{a}z$ are equivalent, thereby allowing the above τ-transitions to be applied one after another. By making this flawed design decision, we would break the contract on the first private input. In particular, the fresh name z unified with x must be guaranteed freshness in the scope of x.

Fortunately, $\nu z.(a[y] \mid \overline{a}z)$ is not equivalent to $a[y] \mid \nu z.\overline{a}z$, since the labelled transitions system naturally only pushes fresh names outward. Therefore, $\nu z.(a[y] \mid \overline{a}z)$ is deadlocked since $\overline{a}z$

cannot guarantee the output is private in the relevant scope. Thus processe $\nu z.(a[y] \mid \overline{a}z)$ is not mutually similar to $a[y] \mid \nu z.\overline{a}z$. Although, we do have that $\nu z.(a[y] \mid \overline{a}z) \preceq a[y] \mid \nu z.\overline{a}z$ holds, established by extending the results of this paper to a combined $\pi/\pi I$-calculus.

7. Conclusion

This work presents a purely logical embedding of π-calculus and πI-calculus processes as formulae in a proof system. The main result is the soundness of linear implication in the logical system with respect to a notion of weak simulation called complete weak open simulation (Theorem 3.8). The proof is constructive, so such weak simulations can be constructed from linear implications. This result sharpens evidence that linear implication *objectively* defines a preorder over processes, preserved in all contexts, in the sense that the semantics of linear implication are determined by the fundamental proof theoretic principal of cut elimination, rather than human design. The result is as a critical step in the roadmap towards situating properly this natural preorder in the spectrum of preorders over processes. In particular, linear implication is a branching-time preorder preserved in all contexts.

The paper also casts light on proof search in the calculus of structures. In Proposition 4.11, we show, for a significant fragment of BV and BV1 (larger than previously considered), "left proofs" can always be constructed, where interaction rules are applied only in certain "left contexts". This normalisation procedure ensures that the leftmost atom in a formula is involved in the bottommost instance of an interaction rule. This suggests that, as for other proof systems Andreoli (1992); Miller et al. (1991), there are significant useful fragments of the calculus of structures for which we can achieve finer control of proof search.

Future work is to pursue the roadmap laid out at the beginning of Section 2. We aim to tighten relationships between linear implication and established preorders over processes; pursuing the hypothesis that (complete weak) variants of ST-simulation, van Glabbeek and Vaandrager (1987); van Glabbeek and Goltz (2001), provide the tightest match for linear implication in the literature. Also, we are in the process of further extending BV such that more expressive process calculi can be embedded, e.g., featuring probabilistic choice. Indeed, one of the advantages of having established that linear implication is a branching-time preorder, is that we know linear implication also respects probabilistic testing, Deng et al. (2007).

Acknowledgements. The authors receive support from MOE, Singapore, Tier 2 grant MOE2014-T2-2-076 and the National Research Foundation Singapore under its National Cybersecurity R&D Program (Award No. NRF2014NCR-NCR001-30). We are grateful to Alessio Guglielmi, who suggested trying a de Morgan dual pair of nominal quantifiers.

References

Abramsky, S. (1994). Proofs as processes, *Theor. Comput. Sci.* **135**(1): 5–9.

Ahn, K. Y., Horne, R. and Tiu, A. (2017). A Characterisation of Open Bisimilarity using an Intuitionistic Modal Logic, *in* R. Meyer and U. Nestmann (eds), *28th International Conference on Concurrency Theory (CONCUR 2017)*, Vol. 85 of *Leibniz International Proceedings in Informatics (LIPIcs)*, Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, pp. 7:1–7:17.

Andreoli, J.-M. (1992). Logic programming with focusing proofs in linear logic, *Journal of Logic and Computation* **2**(3): 297–347.

Bellin, G. and Scott, P. J. (1994). On the pi-calculus and linear logic, *Theor. Comput. Sci.* **135**(1): 11–65.

Bengtson, J. and Parrow, J. (2009). Formalising the pi-calculus using nominal logic, *Logical Methods in Computer Science* **5**(2).

Bernardi, G. and Hennessy, M. (2013). Mutually testing processes, *International Conference on Concurrency Theory*, pp. 61–75.

Brünnler, K. and Tiu, A. F. (2001). A local system for classical logic, *in* R. Nieuwenhuis and A. Voronkov (eds), *Logic for Programming, Artificial Intelligence, and Reasoning*, Lecture Notes in Computer Science, pp. 347–361.

Bruscoli, P. (2002). A purely logical account of sequentiality in proof search, *Logic Programming, 18th International Conference, ICLP 2002, Copenhagen, Denmark, July 29 - August 1, Proceedings*, Vol. 2401 of *Lecture Notes in Computer Science*, pp. 302–316.

Caires, L., Pfenning, F. and Toninho, B. (2016). Linear logic propositions as session types, *Mathematical Structures in Computer Science* **26**(3): 367–423.

Chaudhuri, K., Guenot, N. and Straßburger, L. (2011). The Focused Calculus of Structures, *in* M. Bezem (ed.), *Computer Science Logic (CSL'11) - 25th International Workshop/20th Annual Conference of the EACSL*, Vol. 12 of *Leibniz International Proceedings in Informatics*, Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, pp. 159–173.

Ciobanu, G. and Horne, R. (2015). Behavioural analysis of sessions using the calculus of structures, *in* M. Mazzara and A. Voronkov (eds), *Perspectives of System Informatics*, Springer International Publishing, pp. 91–106.

Deng, Y., Van Glabbeek, R., Hennessy, M., Morgan, C. and Zhang, C. (2007). Characterising testing preorders for finite probabilistic processes, *22nd Annual IEEE Symposium on Logic in Computer Science. LICS 2007*, IEEE, pp. 313–325.

Deniélou, P. and Yoshida, N. (2013). Multiparty compatibility in communicating automata: Characterisation and synthesis of global session types, *Automata, Languages, and Programming - 40th International Colloquium, ICALP 2013, Riga, Latvia, July 8-12*, pp. 174–186.

Gacek, A., Miller, D. and Nadathur, G. (2011). Nominal abstraction, *Information and Computation* **209**(1): 48–73.

Girard, J.-Y. (1987). Linear logic, *Theoretical Computer Science* **50**(1): 1–112.

Gischer, J. L. (1988). The equational theory of pomsets, *Theor. Comput. Sci.* **61**: 199–224.

Guglielmi, A. (2007). A system of interaction and structure, *ACM Transactions on Computational Logic* **8**(1).

Guglielmi, A. and Straßburger, L. (2001). Non-commutativity and MELL in the calculus of structures, *in* L. Fribourg (ed.), *Computer Science Logic*, Lecture Notes in Computer Science, pp. 54–68.

Guglielmi, A. and Straßburger, L. (2011). A system of interaction and structure V: The exponentials and splitting, *Math. Struct. Comp. Sci.* **21**(03): 563–584.

Horne, R. (2015). The consistency and complexity of multiplicative additive system virtual, *Sci. Ann. Comp. Sci.* **25**(2): 245–316.

Horne, R., Mauw, S. and Tiu, A. (2017). Semantics for specialising attack trees based on linear logic, *Fundamenta Informaticae* **153**: 57–86.

Horne, R., Tiu, A., Aman, B. and Ciobanu, G. (2016). Private Names in Non-Commutative Logic, *in* J. Desharnais and R. Jagadeesan (eds), *27th International Conference on Concurrency Theory (CONCUR 2016)*, Vol. 59 of *Leibniz International Proceedings in Informatics (LIPIcs)*, Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, pp. 31:1–31:16.

Horne, R., Tiu, A., Aman, B. and Ciobanu, G. (2018). Polarised nominal quantifiers model private names in non-commutative logic, *Technical Report 1502*. ISSN 1842-1490, extended version of above supporting submission to TOCL.
URL: *http://iit.iit.tuiasi.ro/TR/reports/fml1502.pdf*

McDowell, R., Miller, D. and Palamidessi, C. (2003). Encoding transition systems in sequent calculus, *Theor. Comput. Sci.* **294**(3): 411–437.

Miller, D. (1993). The pi-calculus as a theory in linear logic: Preliminary results, *Extensions of Logic Programming, Third International Workshop, ELP'92, Bologna, Italy, February 26-28, 1992, Proceedings*, Vol. 660 of *Lecture Notes in Computer Science*, pp. 242–264.

Miller, D., Nadathur, G., Pfenning, F. and Scedrov, A. (1991). Uniform proofs as a foundation for logic programming, *Annals of Pure and Applied Logic* **51**(1): 125 – 157.

Miller, D. and Tiu, A. (2005). A proof theory for generic judgements, *ACM Transactions on Computational Logic (TOCL)* **6**(4): 749–783.

Milner, R. (1989). *Communication and Concurrency*, Prentice-Hall International.

Milner, R., Parrow, J. and Walker, D. (1992). A calculus of mobile processes, I and II, *Information and Computation* **100**(1): 1–77.

Pitts, A. (2003). Nominal logic, a first order theory of names and binding, *Information and Computation* **186**(2).

Retoré, C. (1997). Pomset logic: A non-commutative extension of classical linear logic, *in* P. de Groote and J. Roger Hindley (eds), *Typed Lambda Calculi and Applications*, Lecture Notes in Computer Science, pp. 300–318.

Roversi, L. (2016). A deep inference system with a self-dual binder which is complete for linear lambda calculus, *Journal of Logic and Computation* **26**(2): 677.

Sangiorgi, D. (1996a). π-calculus, internal mobility, and agent-passing calculi, *Theoretical Computer Science* **167**(1): 235–274.

Sangiorgi, D. (1996b). A theory of bisimulation for the π-calculus, *Acta inform.* **33**(1): 69–97.

Sassone, V., Nielsen, M. and Winskel, G. (1996). Models for concurrency: towards a classification, *Th. Comp. Sci.* **170**(1-2): 297–348.

Straßburger, L. and Guglielmi, A. (2011). A system of interaction and structure IV: the exponentials and decomposition, *ACM Transactions on Computational Logic (TOCL)* **12**(4): 23.

Tiu, A. (2006). A system of interaction and structure II: The need for deep inference, *Logical Methods in Computer Science* **2**(2:4): 1–24.

Tiu, A. and Miller, D. (2010). Proof search specifications of bisimulation and modal logics for the π-calculus, *ACM Transactions on Computational Logic* **11**(2): 13.

van Glabbeek, R. (1990). The linear time-branching time spectrum (extended abstract), *CONCUR '90, Amsterdam, The Netherlands, August 27-30*, Vol. 458 of *Lecture Notes in Computer Science*, pp. 278–297.

van Glabbeek, R. and Goltz, U. (2001). Refinement of actions and equivalence notions for concurrent systems, *Acta Informatica* **37**(4-5): 229–327.

van Glabbeek, R. and Vaandrager, F. (1987). Petri net models for algebraic theories of concurrency, *in* J. W. de Bakker, A. J. Nijman and P. C. Treleaven (eds), *PARLE Parallel Architectures and Languages Europe*, pp. 224–242.