BV and Pomset Logic Are Not the Same

Lê Thành Dũng (Tito) Nguyễn
CNRS & IRISA, Rennes, France

Lutz Straßburger
Inria Saclay & École Polytechnique, Palaiseau, France

Abstract

BV and pomset logic are two logics that both conservatively extend unit-free multiplicative linear logic by a third binary connective, which (i) is non-commutative, (ii) is self-dual, and (iii) lies between the "par" and the "tensor". It was conjectured early on (more than 20 years ago), that these two logics, that share the same language, that both admit cut elimination, and whose connectives have essentially the same properties, are in fact the same. In this paper we show that this is not the case. We present a formula that is provable in pomset logic but not in BV.

2012 ACM Subject Classification Theory of computation → Linear logic; Theory of computation → Proof theory

Keywords and phrases proof nets, deep inference, pomset logic, system BV, cographs, dicographs, series-parallel orders

Digital Object Identifier 10.4230/LIPIcs.CSL.2022.32

1 Introduction

Pomset logic has been discovered by Christian Retoré [21] through the study of coherence spaces which form a semantics of proofs for linear logic. Retoré observed that next to the two operations $\otimes$ (*tensor* or *multiplicative conjunction*) and $\parr$ (*par* or *multiplicative disjunction*) there are two other operations $\triangleleft$ and $\triangleright$, which are non-commutative, obey $A \triangleleft B = B \triangleright A$, and are self-dual, i.e., $\langle A \triangleleft B \rangle^\perp = A^\perp \triangleleft B^\perp$.[1] From this semantic observation, Retoré derived a proof net syntax together with a correctness criterion and a cut elimination theorem. However, he could not provide a sound and complete cut-free sequent calculus for this logic [20]. Nonetheless, pomset logic has found applications in linguistics, as basis of a new categorial grammar [17], similar to the ones based on the Lambek calculus [16].

System BV was found by Alessio Guglielmi [10] through a syntactic investigation of the connectives of pomset logic and a graph theoretic study of series-parallel orders and cographs. The difficulty of presenting this combination of commutative and non-commutative connectives in the sequent calculus triggered the development of the *calculus of structures* [11], the first proper deep inference proof formalism[2]. The mixture of commutative and non-commutative connectives in BV immediately found applications in computer science, in particular, Bruscoli [3] established a strict correspondence between the proof-search space of BV and the computations in a fragment of CCS. This work was later extended by quantifiers to capture private names and to establish a correspondence of implication in (first-order) BV and a form of weak bisimulation in the π-calculus [12, 13].

---

[1] Observe that the order is *not* inverted, as it is the case with other non-commutative variants of linear logic [29] (see also [9, Section II.9.]).

[2] The basic idea of such a rewriting system goes back to Retoré [22] (see also [4]), but not as a proof system admitting cut-elimination.

30th EACSL Annual Conference on Computer Science Logic (CSL 2022).
Editors: Florin Manea and Alex Simpson; Article No. 32; pp. 32:1–32:17
Leibniz International Proceedings in Informatics
LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

This leads to the strange situation that we have two logics, pomset logic and BV, which are both conservative extensions of unit-free multiplicative linear logic with mix (MLL_0) [8, 7] with a non-commutative connective $\lhd$ such that $A \otimes B \multimap A \lhd B \multimap A \parr B$, which both obey a cut elimination result, and which both have found applications that lie outside of pure proof theory.

The only difference between the two logics is that pomset logic naturally extends the proof net correctness criterion of MLL_0 to the new non-commutative connective, but has no deductive proof system, whereas BV naturally extends a deductive system for MLL_0 with the new non-commutative connective, but has no proof nets. This naturally led to the conjecture that both logics ought to be the same [28]. In fact, most researchers working in this area (including the second author of this paper) believed that the two logics comprise the same set of theorems.

In this paper we show that this is not the case. More precisely, we show that the theorems of BV form a proper subset of the theorems of pomset logic. It has already been observed before [22, 28, 26] that every theorem in BV is also a theorem of pomset logic. However, the converse is not true, and we give an example of a formula that is a theorem of pomset logic but not provable in BV.

Organisation of this paper

In the next two sections we give some preliminaries on pomset logic (Section 2) and BV (Section 3). Then, in Section 4 we show that BV is contained in pomset logic. Even though this has been known since more than 20 years [22, 28], there has been no complete proof published so far. The proof we present here is a simplification of the one suggested in [28]. Next, in Section 5, we give our counterexample showing that the converse is not true, i.e., we present a formula that is a theorem of pomset logic but not provable in BV. Finally, in the conclusion (Section 6), we discuss some complexity results and give some intuition on how the counterexample has been found and why it took so long to find it.

2 Preliminaries on Pomset Logic

The **formulas** of pomset logic and BV are in this paper denoted by capital Latin letters $A, B, C, \ldots$ and are generated from a countable set $\mathcal{V} = \{a, b, c, \ldots\}$ of propositional variables and the **unit** $\mathbb{I}$ via the three binary connectives **tensor** $\otimes$, **par** $\parr$, and **seq** $\lhd$, according to the grammar

$$A, B \quad ::= \quad \mathbb{I} \mid a \mid a^\perp \mid (A \otimes B) \mid [A \parr B] \mid \langle A \lhd B \rangle \tag{1}$$

An **atom** is either a propositional variable or its dual. For a formula A, we define its **size** $|A|$ to be the number of atom occurrences in A. For better readability of large formulas, we use here different kinds of parentheses for the different connectives.[3] In the following, we omit outermost parentheses for better readability. The unit $\mathbb{I}$ behaves as unit for all three connectives. We define the *relation* $\equiv$ *on formulas* to be the smallest congruence generated by associativity of $\otimes, \parr, \lhd$, commutativity of $\otimes, \parr$, and the unit equations:

$$\begin{array}{rcc} A \otimes (B \otimes C) \equiv (A \otimes B) \otimes C & A \otimes B \equiv B \otimes A & \mathbb{I} \otimes A \equiv A \\ A \parr [B \parr C] \equiv [A \parr B] \parr C & A \parr B \equiv B \parr A & \mathbb{I} \parr A \equiv A \\ A \lhd \langle B \lhd C \rangle \equiv \langle A \lhd B \rangle \lhd C & & \mathbb{I} \lhd A \equiv A \equiv A \lhd \mathbb{I} \end{array} \tag{2}$$

---

[3] Note that this is redundant and carries no additional meaning. The only purpose is better readability.

The involutive *(linear) negation* $(-)^\perp$ is extended from propositional variables to general formulas by taking De Morgan's laws as its inductive definition, i.e., we define $(a^\perp)^\perp = a$ for all propositional variables a, and

$$\mathbb{I}^\perp = \mathbb{I} \qquad (A \otimes B)^\perp = A^\perp \parr B^\perp \qquad [A \parr B]^\perp = A^\perp \otimes B^\perp \qquad \langle A \lhd B\rangle^\perp = A^\perp \lhd B^\perp$$

The last equality is what we mean when we say that seq is *self-dual*. Note that the right-hand side is indeed $A^\perp \lhd B^\perp$ and not $B^\perp \lhd A^\perp$.[4]

We will also need the notion of **sequent**, which has to be generalized from multisets of formulas to series-parallel orders of formulas.[5] We denote a *sequent* in pomset logic by capital Greek letters $\Gamma, \Delta, \ldots$ and they are generated as follows: $\Gamma, \Delta ::= \varnothing \mid A \mid [\Gamma, \Delta] \mid \langle \Gamma; \Delta\rangle$, where $\varnothing$ stands for the empty sequent. We consider sequents equal modulo commutativity of $[\cdot,\cdot]$ and associativity of $[\cdot,\cdot]$ and $\langle\cdot;\cdot\rangle$, and the unit-laws for the empty sequent. In the remainder of this paper we will always omit redundant brackets.

The operations $[\cdot,\cdot]$ and $\langle\cdot;\cdot\rangle$ serve as counterparts on sequents to the connectives $\parr$ and $\lhd$ on formulas (just as the sequent $\vdash A, B, C$ morally means $A \parr B \parr C$ in linear logic).

▶ Remark 2.1. Pomset logic is not the only system that features "non-flat" sequents with two distinct connectives. Another famous example is the logic BI of bunched implications [19].

In [21], Retoré presents proof nets for pomset logic as *RB-digraphs*, that is, *directed* graphs equipped with perfect matchings, extending his reformulation of MLL_0 proof nets as *undirected* RB-graphs [23]. We recall these notions below.

▶ **Definition 2.2.** *A **digraph** $\mathcal{G} = (V_\mathcal{G}, E_\mathcal{G})$ consists of a finite set of **vertices** $V_\mathcal{G}$ and a set of **edges** $E_\mathcal{G} \subseteq V_\mathcal{G}^2 \setminus \{(u,u) \mid u \in V_\mathcal{G}\}$. A digraph $\mathcal{G}$ is **labeled** if there is a map $\ell \colon V_\mathcal{G} \to \mathcal{L}$ assigning each vertex v of $V_\mathcal{G}$ a **label** $\ell(v) \in \mathcal{L}$ in the **label set** $\mathcal{L}$. If $\mathcal{L}$ is the set $\mathcal{V} \cup \mathcal{V}^\perp$ of atoms, we speak of an **atom-labeled** digraph.*

In the remainder of this paper, all digraphs are atom-labelled, and for two digraphs $\mathcal{G}$ and $\mathcal{H}$, we write $\mathcal{G} = \mathcal{H}$ iff there is a label-preserving isomophism between them. Also, we often write $uv \in E_\mathcal{G}$ for $(u,v) \in E_\mathcal{G}$, and for a digraph $\mathcal{G} = (V_\mathcal{G}, E_\mathcal{G})$, we define the sets $E_\mathcal{G}^\otimes = \{(u,v) \mid (u,v) \in E_\mathcal{G} \text{ and } (v,u) \in E_\mathcal{G}\}$ and $E_\mathcal{G}^\lhd = \{(u,v) \mid (u,v) \in E_\mathcal{G} \text{ and } (v,u) \notin E_\mathcal{G}\}$, allowing us to treat $(V_\mathcal{G}, E_\mathcal{G}^\otimes)$ as undirected graph.

▶ **Definition 2.3.** *Let $\mathcal{G} = (V_\mathcal{G}, E_\mathcal{G})$ and $\mathcal{H} = (V_\mathcal{H}, E_\mathcal{H})$ be disjoint digraphs. We can define the following operations:*

$$\begin{aligned}
\mathcal{G} \parr \mathcal{H} &= (V_\mathcal{G} \cup V_\mathcal{H}, E_\mathcal{G} \cup E_\mathcal{H}) \\
\mathcal{G} \lhd \mathcal{H} &= (V_\mathcal{G} \cup V_\mathcal{H}, E_\mathcal{G} \cup E_\mathcal{H} \cup \{(u,v) \mid u \in V_\mathcal{G} \text{ and } v \in V_\mathcal{H}\}) \\
\mathcal{G} \otimes \mathcal{H} &= (V_\mathcal{G} \cup V_\mathcal{H}, E_\mathcal{G} \cup E_\mathcal{H} \cup \{(u,v),(v,u) \mid u \in V_\mathcal{G} \text{ and } v \in V_\mathcal{H}\})
\end{aligned}$$

This allows us to define a mapping $[\![\cdot]\!]$ from formulas to digraphs as follows:

$$[\![\mathbb{I}]\!] = \varnothing \qquad [\![a]\!] = \bullet_a \qquad [\![a^\perp]\!] = \bullet_{a^\perp}$$

$$[\![A \parr B]\!] = [\![A]\!] \parr [\![B]\!] \qquad [\![A \lhd B]\!] = [\![A]\!] \lhd [\![B]\!] \qquad [\![A \otimes B]\!] = [\![A]\!] \otimes [\![B]\!]$$

where $\varnothing$ is the empty graph, and $\bullet_a$ (respectively $\bullet_{a^\perp}$) is a single vertex graph whose vertex is labeled by a (respectively $a^\perp$).

---

[4] In that respect, pomset logic and BV are different from other non-commutative variants of linear logic where $\otimes$ and $\parr$ are non-commutative with $(A \otimes B)^\perp = B^\perp \parr A^\perp$ [29, 1].

[5] We follow here mainly the presentation of [24].

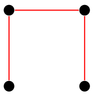
▶ **Proposition 2.4** ([22]). *For all formulas A and B, we have $[\![A]\!] = [\![B]\!]$ iff $A \equiv B$.*

This can be shown by a straightforward induction on the formulas. An immediate consequence of this proposition is that the extension of the mapping $[\![\cdot]\!]$ to sequents is well-defined, i.e., we have $[\![\Gamma, \Delta]\!] = [\![\Gamma]\!] \parr [\![\Delta]\!]$ and $[\![\Gamma ; \Delta]\!] = [\![\Gamma]\!] \triangleleft [\![\Delta]\!]$.

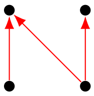
▶ **Definition 2.5.** *Let $\mathcal{G} = (V_\mathcal{G}, E_\mathcal{G})$ be a digraph and let $V_\mathcal{H} \subseteq V_\mathcal{G}$. The **subdigraph** of $\mathcal{G}$ **induced by** $V_\mathcal{H}$ is $\mathcal{H} = (V_\mathcal{H}, E_\mathcal{H})$, where $E_\mathcal{H} = \{(u, v) \mid (u, v) \in E_\mathcal{G}$ and $u \in V_\mathcal{H}$ and $v \in V_\mathcal{H}\}$. In this case we also say that $\mathcal{H}$ is an **induced subgraph** of $\mathcal{G}$ and denote that by $\mathcal{H} \sqsubseteq \mathcal{G}$. If additionally $V_\mathcal{H} \subset V_\mathcal{G}$ then we write $\mathcal{H} \sqsubset \mathcal{G}$.*

▶ **Definition 2.6.** *An undirected graph is $\mathbf{P_4}$**-free** if it does not contain a $\mathbf{P_4}$ (shown on the left below) as induced subgraph, and a directed graph is $\mathbf{N}$**-free** if it does not contain an $\mathbf{N}$ (shown on the right below) as induced subgraph.*

$\mathbf{P_4}$:



$\mathbf{N}$:



(3)

▶ **Definition 2.7.** *A **dicograph** is a digraph $\mathcal{G} = (V_\mathcal{G}, E_\mathcal{G})$, such that*

1. *the undirected graph $(V_\mathcal{G}, E_\mathcal{G}^{\otimes})$ is $\mathbf{P_4}$-free,*
2. *the directed graph $(V_\mathcal{G}, E_\mathcal{G}^{\triangleleft})$ is $\mathbf{N}$-free, and*
3. *the relation $E_\mathcal{G}$ is **weakly transitive**:*
   - *if $(u, v) \in E_\mathcal{G}^{\triangleleft}$ and $(v, w) \in E_\mathcal{G}$ then $(u, w) \in E_\mathcal{G}$, and*
   - *if $(u, v) \in E_\mathcal{G}$ and $(v, w) \in E_\mathcal{G}^{\triangleleft}$ then $(u, w) \in E_\mathcal{G}$.*

▶ **Proposition 2.8** ([4]). *$\mathcal{G}$ is a dicograph iff there is a formula A with $\mathcal{G} = [\![A]\!]$.*

▶ **Proposition 2.9.** *Let $\mathcal{G} = (V_\mathcal{G}, E_\mathcal{G})$ be a dicograph. Then any induced subdigraph of $\mathcal{G}$ is also a dicograph.*

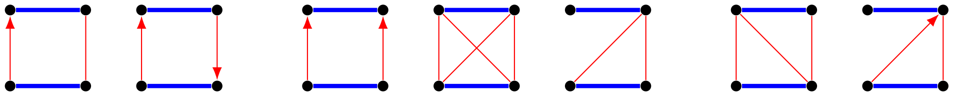
▶ **Definition 2.10.** *Let $\mathcal{G} = (V_\mathcal{G}, E_\mathcal{G})$ be a digraph. A **perfect matching** B of $\mathcal{G}$ is a subset of edges such that:*

1. *any vertex has exactly one outgoing edge in B and exactly one incoming edge in B, i.e., for every $u \in V_\mathcal{G}$ there is exactly one pair $(v, w) \in V_\mathcal{G} \times V_\mathcal{G}$ such that $uv \in B$ and $wu \in B$, and*
2. *for all $u, v \in V_\mathcal{G}$, we have that $uv \in B$ iff $vu \in B$.*

*Item 2 means that B consists of bidirectional edges. In particular, this means that $v = w$ in Item 1. An **RB-digraph** $\mathcal{G} = (V_\mathcal{G}, R_\mathcal{G}, B_\mathcal{G})$ is a triple where $(V_\mathcal{G}, R_\mathcal{G} \uplus B_\mathcal{G})$ is a digraph and $B_\mathcal{G}$ is a perfect matching in it. Finally, an RB-digraph $\mathcal{G} = (V_\mathcal{G}, R_\mathcal{G}, B_\mathcal{G})$ is an **RB-dicograph** iff $(V_\mathcal{G}, R_\mathcal{G})$ is a dicograph.*[6]

In all figures representing RB-digraphs, we will (following [22]) draw the edges belonging to the matching (the set B) **bold and blue**, and the other edges (the set R) regular and red.

▶ **Example 2.11.** Below we show 7 examples of RB-digraphs. The first 5 are RB-dicographs, the last 2 are not.



(4)

---

[6] Note that the perfect matching $B_\mathcal{G}$ is not part of the dicograph. In particular, we allow that two vertices in $V_\mathcal{G}$ can be connected by an edge in $R_\mathcal{G}$ and in $B_\mathcal{G}$.

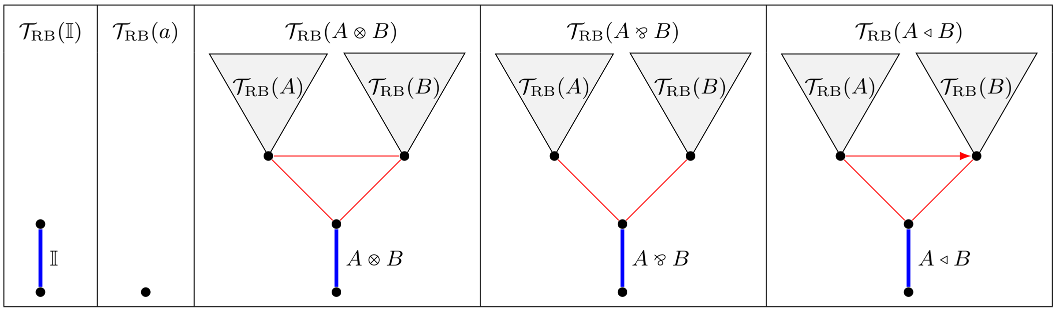


Figure 1 Inductive definition of RB-trees (which are not quite trees in the sense of graph theory, though they resemble the syntax trees of formulas). The root vertex is at the bottom.

▶ **Definition 2.12.** *An **elementary cycle** of length n in a digraph $(V_{\mathcal{G}}, E_{\mathcal{G}})$ is a $\mathbb{Z}/n\mathbb{Z}$-indexed sequence of vertices $u_0, \ldots, u_{n-1} \in V_{\mathcal{G}}$ without repetitions such that for all $i \in \mathbb{Z}/n\mathbb{Z}$, $u_i u_{i+1} \in E_{\mathcal{G}}$. An **alternating elementary cycle** (or **æ-cycle**) in an RB-digraph $(V_{\mathcal{G}}, R_{\mathcal{G}}, B_{\mathcal{G}})$ is an elementary cycle $u_0, \ldots, u_{n-1}$ in $(V_{\mathcal{G}}, R_{\mathcal{G}} \uplus B_{\mathcal{G}})$, such that for all $i \in \mathbb{Z}/n\mathbb{Z}$, exactly one of $u_{i-1}u_i$ and $u_i u_{i+1}$ is in $B_{\mathcal{G}}$ (so that the other one is in $R_{\mathcal{G}}$). Note that this forces the length n of an æ-cycle to be even. A **chord** in an æ-cycle is an edge $vw \in R_{\mathcal{G}}$ such that $v, w \in \{u_0, \ldots, u_{n-1}\}$ but neither vw nor wv are in the æ-cycle. We say an æ-cycle is **chordless** if it does not admit any chord in $\mathcal{G}$. We say that an RB-digraph $(V_{\mathcal{G}}, R_{\mathcal{G}}, B_{\mathcal{G}})$* is *an æ-cycle (resp. chordless æ-cycle) if all vertices of $V_{\mathcal{G}}$ participate in the cycle. Finally, an RB-digraph is **æ-acyclic** if it does not contain a chordless æ-cycle as induced subgraph.*

▶ **Example 2.13.** To continue Example 2.11, the first two RB-digraphs in (4) are chordless æ-cycles. The other five are æ-acyclic.

We are now ready to define pomset logic proof nets, which are in fact æ-acyclic RB-dicographs.

A **pomset logic pre-proof** of a sequent Γ is an involution ℓ on its set of atom occurrences such that an atom is always mapped to its dual. This involutive mapping on the atom occurrences is called the **axiom linking**.

In order to define which pre-proofs are proofs, Retoré [21, 22] gave two equivalent correctness criteria, which are in fact two ways of translating the sequent Γ and the linking ℓ into an RB-dicograph.

Let us call the first the **relational RB-prenet**, denoted by $\rho(\Gamma, \ell)$, which is the RB-dicograph $\mathcal{G} = (V_{\mathcal{G}}, R_{\mathcal{G}}, B_{\mathcal{G}})$ where $(V_{\mathcal{G}}, R_{\mathcal{G}}) = [\![\Gamma]\!]$, and we have $xy \in B_{\mathcal{G}}$ iff the atoms in Γ that correspond to x and y are mapped to each other by the axiom linking ℓ.

▶ **Example 2.14.** The first five RB-graphs in (4) are in fact relational RB-prenets for the formulas $\langle a^\perp \lhd b^\perp \rangle \parr (a \otimes b)$, $\langle a^\perp \lhd b^\perp \rangle \parr \langle b \lhd a \rangle$, and $\langle a^\perp \lhd b^\perp \rangle \parr \langle a \lhd b \rangle$, $[a \parr a^\perp] \otimes [b \parr b^\perp]$, $a \parr (a^\perp \otimes [b \parr b^\perp])$, respectively (with the obvious unique linking).

The second way of translating a sequent Γ and a linking ℓ into an RB-dicograph is based on the formula tree structure. We define inductively for each formula C in Γ its **RB-tree**, denoted as $\mathcal{T}_{\mathrm{RB}}(C)$, as shown in Figure 1.[7]

---

[7] Technically speaking, this not a tree in the graph-theoretical sense, but we use the name as it carries the structure of the formula tree.

$$\mathsf{ai}{\downarrow}\,\frac{\mathbb{I}}{a \parr a^{\perp}} \qquad \mathsf{s}\,\frac{[A \parr C] \otimes B}{(A \otimes B) \parr C} \qquad \mathsf{q}{\downarrow}\,\frac{[A \parr C] \lhd [B \parr D]}{\langle A \lhd B\rangle \parr \langle C \lhd D\rangle} \qquad \equiv\,\frac{A}{B}\ \text{(provided } A{\equiv}B\text{)}$$

Figure 2 System BV.

If we have a sequent Γ, then $\mathcal{T}_{\mathrm{RB}}(\Gamma)$ is obtained from the RB-trees of the formulas in Γ which are connected at the roots via the edges corresponding to the series-parallel order of the sequent structure. In order to obtain an RB-digraph, we need to add the B-edges corresponding to the linking ℓ. We denote this RB-digraph, which is in fact an RB-dicograph, by $\tau(\Gamma, \ell)$ and call it the **tree-like RB-prenet** of Γ and ℓ.

▶ **Definition 2.15.** *A relational RB-prenet (resp. tree-like RB-prenet) is* ***correct*** *if it does not contain any chordless æ-cycle. A correct relational RB-prenet (resp. correct tree-like RB-prenet) is also called a* ***relational RB-net*** *(resp.* ***tree-like RB-net****). In both cases we also speak of* ***(pomset logic) proof nets****. A sequent Γ is provable in pomset logic of there is a linking ℓ, such that $\rho(\Gamma, \ell)$ or $\tau(\Gamma, \ell)$ is a proof net.*

The above definition makes sense because of the following theorem by Retoré:

▶ **Theorem 2.16** ([22, Theorem 7])**.** *For every sequent Γ and linking ℓ, we have that $\rho(\Gamma, \ell)$ is correct if and only if $\tau(\Gamma, \ell)$ is correct.*

▶ **Example 2.17.** The three RB-graphs in the middle of (4) are pomset logic proof nets.

3 Preliminaries on System BV

In [10] Guglielmi introduces **system** BV, which is a deductive system for formulas defined in (1). It is defined in the formalism called the *calculus of structures*, and it works similar to a rewriting system, modulo the equational theory defined in (2).

The inference rules of **system** BV are shown in Figure 2. These rules have to be read as rewriting rule schemes, meaning that (i) the variable a can be substituted by any atom, and the variables A, B, C, D can be substituted by any formula, and that (ii) the rules can be applied inside any (positive) context.

A **(proof) system** is a set of inference rules. We write $\begin{array}{c}A\\ \mathsf{S}\,\Vert\,\delta\\ B\end{array}$, or more concisely $A \vdash^{\delta}_{\mathsf{S}} B$, if there is a derivation from A to B using only rules from the system S, and that derivation is named δ. If in that situation $A = \mathbb{I}$, then we write it as $\begin{array}{c}\mathsf{S}\,\Vert\,\delta\\ B\end{array}$ or simply as $\vdash^{\delta}_{\mathsf{S}} B$ and call δ a **proof** of B. In this case we say that B is **provable** S.

▶ **Example 3.1.** Here are three proofs in BV, corresponding to the three proof nets in the middle of (4):

$$\mathsf{q}{\downarrow}\,\cfrac{\mathsf{ai}{\downarrow}\,\cfrac{\equiv\,\cfrac{\mathsf{ai}{\downarrow}\,\cfrac{\mathbb{I}}{b^{\perp} \parr b}}{\mathbb{I} \lhd [b^{\perp} \parr b]}}{[a^{\perp} \parr a] \lhd [b^{\perp} \parr b]}}{\langle a^{\perp} \lhd b^{\perp}\rangle \parr \langle a \lhd b\rangle} \qquad \mathsf{ai}{\downarrow}\,\cfrac{\mathsf{ai}{\downarrow}\,\cfrac{\equiv\,\cfrac{\mathbb{I}}{\mathbb{I} \otimes \mathbb{I}}}{\mathbb{I} \otimes [b \parr b^{\perp}]}}{[a \parr a^{\perp}] \otimes [b \parr b^{\perp}]} \qquad \mathsf{ai}{\downarrow}\,\cfrac{\equiv\,\cfrac{\mathsf{ai}{\downarrow}\,\cfrac{\mathbb{I}}{a \parr a^{\perp}}}{a \parr (a^{\perp} \otimes \mathbb{I})}}{a \parr (a^{\perp} \otimes [b \parr b^{\perp}])} \tag{5}$$

$$
\begin{array}{lll}
\mathsf{ai}^{\circ}\!\downarrow\dfrac{}{a \parr a^{\perp}} & \mathsf{ai}^{\otimes}\!\downarrow\dfrac{B}{[a \parr a^{\perp}] \otimes B} & \equiv'\dfrac{A}{B}\ (\text{provided } A \equiv' B) \\[2ex]
\mathsf{ai}^{\triangleleft}_{\mathsf{L}}\!\downarrow\dfrac{B}{[a \parr a^{\perp}] \triangleleft B} & \mathsf{ai}^{\triangleleft}_{\mathsf{R}}\!\downarrow\dfrac{B}{B \triangleleft [a \parr a^{\perp}]} & \\[2ex]
\mathsf{q}^{\mathsf{L}}_{3}\!\downarrow\dfrac{[A \parr C] \triangleleft B}{\langle A \triangleleft B\rangle \parr C} & \mathsf{q}^{\mathsf{R}}_{3}\!\downarrow\dfrac{A \triangleleft [B \parr C]}{\langle A \triangleleft B\rangle \parr C} & \mathsf{s}_3\dfrac{[A \parr C] \otimes B}{(A \otimes B) \parr C} \\[2ex]
\mathsf{q}_4\!\downarrow\dfrac{[A \parr C] \triangleleft [B \parr D]}{\langle A \triangleleft B\rangle \parr \langle C \triangleleft D\rangle} & \mathsf{q}_2\!\downarrow\dfrac{A \triangleleft B}{A \parr B} & \mathsf{s}_2\dfrac{A \otimes B}{A \parr B}
\end{array}
$$

Figure 3 System BVu.

An inference rule r is **derivable** in a system S iff for every instance $\mathsf{r}\dfrac{A}{B}$ there is a derivation $A \vdash_{\mathsf{S}} B$. An inference rule r is **admissible** for a system S iff for every proof $\vdash_{\mathsf{S}\cup\{\mathsf{r}\}} A$ there is a proof $\vdash_{\mathsf{S}} B$.

▶ **Definition 3.2.** *Two system* S_1 *and* S_2 *are* ***equivalent*** *if they prove the same formulas.*

To simplify the proofs of our main results, we need a unit-free version of BV. We use here a variant of the one proposed by Kahramanoğulları in [14] in order to reduce the non-determinism in proof search in BV.

The system is called BVu, and its formulas are the same as defined in (1), except that we do not allow any occurrence of the unit $\mathbb{I}$. This means that we have to restrict the equivalence $\equiv$ defined in (2) to the unit-free formulas. We define the relation $\equiv'$ to be the smallest congruence generated by

$$
\begin{array}{rcl@{\qquad\qquad}rcl}
A \otimes (B \otimes C) & \equiv' & (A \otimes B) \otimes C & A \otimes B & \equiv' & B \otimes A \\
A \parr [B \parr C] & \equiv' & [A \parr B] \parr C & A \parr B & \equiv' & B \parr A \\
A \triangleleft \langle B \triangleleft C\rangle & \equiv' & \langle A \triangleleft B\rangle \triangleleft C & & &
\end{array}
\tag{6}
$$

The inference rules for BVu are then shown in Figure 3.[8] Note that the rule $\mathsf{ai}^{\circ}\!\downarrow$ has no premise. It is an axiom that is used exactly once in a **proof** which is a derivation without premise (as the unit $\mathbb{I}$ is not present and cannot take this role).

▶ **Proposition 3.3** ([14])**.** *The systems* BVu *and* BV *are equivalent.*

Proof. First, if we have a proof $\vdash_{\mathsf{BVu}} A$ then we can simply replace the top instance of $\mathsf{ai}^{\circ}\!\downarrow$ by $\mathsf{ai}\!\downarrow$ and have a proof of BV. Conversely, assume we have a proof $\vdash^{\delta}_{\mathsf{BV}} B$. Then, in δ, the unit $\mathbb{I}$ can occur. Let δ' be obtained from δ by deleting the unit $\mathbb{I}$ everywhere (which means that the topmost $\mathsf{ai}\!\downarrow$ is replaced by $\mathsf{ai}^{\circ}\!\downarrow$). Then every instance of the rule $\equiv$ becomes an instance of $\equiv'$; every instance of $\mathsf{q}\!\downarrow$ becomes an instance of $\mathsf{q}_2\!\downarrow$ or $\mathsf{q}^{\mathsf{L}}_3\!\downarrow$ or $\mathsf{q}^{\mathsf{R}}_3\!\downarrow$ or $\mathsf{q}_4\!\downarrow$ or trivial (i.e., premise and conclusion of the rule instance become equal); and similarly for s. However, an instance of $\mathsf{ai}\!\downarrow$ can become an instance of $\mathsf{ai}^{\otimes}\!\downarrow$ or $\mathsf{ai}^{\triangleleft}_{\mathsf{L}}\!\downarrow$ or $\mathsf{ai}^{\triangleleft}_{\mathsf{R}}\!\downarrow$ (which are in BVu), or $\mathsf{ai}^{\parr}\!\downarrow$ which is shown below.

$$
\mathsf{ai}^{\parr}\!\downarrow\dfrac{B}{a \parr a^{\perp} \parr B}
\tag{7}
$$

This rule is not in BVu, but can be derived with $\{\mathsf{ai}^{\otimes}\!\downarrow, \mathsf{s}_2\}$. ◀

---

[8] The rules in the bottom two rows of Figure 3 have have already been studied by Retoré in [22], as part of a rewrite system on digraphs to generate theorems of pomset logic.

▶ Remark 3.4. Our version of BVu is slightly different from the one by Kahramanoğulları [14]. In [14] the rule s_2 is absent, and instead the rule $\mathsf{ai}^{\parr}{\downarrow}$ shown in (7) is part of the system. It is easy to see that the two variants of BVu are equivalent: first, as we have mentioned above, the rule $\mathsf{ai}^{\parr}{\downarrow}$ is derivable in $\{\mathsf{ai}^{\otimes}{\downarrow}, \mathsf{s}_2\}$, and second, the rule s_2 is admissible if $\mathsf{ai}^{\parr}{\downarrow}$ is present. This can be seen by an easy induction on the size of the derivation. However, note that the same trick does not work for the rule $\mathsf{q}_2{\downarrow}$. This rule cannot be shown admissible, as the formula $\langle a \triangleleft [b \parr c]\rangle \parr \langle [a^\perp \parr b^\perp] \triangleleft c^\perp\rangle$ is not provable in BVu without $\mathsf{q}_2{\downarrow}$.

We will also need a variant of BVu that we call $\mathsf{BV\hat{u}}$ and that is obtained from BVu by restricting rules $\mathsf{q}_2{\downarrow}$ and s_2 to cases where neither A nor B has a $\parr$ as main connective, i.e., we replace $\mathsf{q}_2{\downarrow}$ and s_2 by $\hat{\mathsf{q}}_2{\downarrow}$ and $\hat{\mathsf{s}}_2$, respectively:

$$\hat{\mathsf{q}}_2{\downarrow}\,\frac{A \triangleleft B}{A \parr B} \qquad \hat{\mathsf{s}}_2\,\frac{A \otimes B}{A \parr B} \qquad \text{where } A \not\equiv' C \parr D \text{ and } B \not\equiv' C \parr D \text{ for any formulas } C \text{ and } D. \tag{8}$$

and similarly, by restricting the rules $\mathsf{q}_3^{\mathsf{L}}{\downarrow}$, $\mathsf{q}_3^{\mathsf{R}}{\downarrow}$, and s_3 to cases where C does not have a $\parr$ as main connective, i.e., these three rules are replaced by $\hat{\mathsf{q}}_3^{\mathsf{L}}{\downarrow}$, $\hat{\mathsf{q}}_3^{\mathsf{R}}{\downarrow}$, and $\hat{\mathsf{s}}_3$, respectively:

$$\hat{\mathsf{q}}_3^{\mathsf{L}}{\downarrow}\,\frac{[A \parr C] \triangleleft B}{\langle A \triangleleft B\rangle \parr C} \qquad \hat{\mathsf{q}}_3^{\mathsf{R}}{\downarrow}\,\frac{A \triangleleft [B \parr C]}{\langle A \triangleleft B\rangle \parr C} \qquad \hat{\mathsf{s}}_3\,\frac{[A \parr C] \otimes B}{(A \otimes B) \parr C} \qquad \text{where } C \not\equiv' D \parr E \text{ for any formulas } D \text{ and } E. \tag{9}$$

▶ **Proposition 3.5.** *The systems* BVu *and* $\mathsf{BV\hat{u}}$ *are equivalent.*

Proof. Any derivation in $\mathsf{BV\hat{u}}$ is also a derivation in BVu. Conversely, the rules $\mathsf{q}_2{\downarrow}$ and s_2 and s_3 are derivable with $\{\hat{\mathsf{q}}_2{\downarrow}, \hat{\mathsf{q}}_3^{\mathsf{L}}{\downarrow}, \hat{\mathsf{q}}_3^{\mathsf{R}}{\downarrow}, \equiv'\}$ and $\{\hat{\mathsf{s}}_2, \hat{\mathsf{s}}_3, \equiv'\}$ and $\{\hat{\mathsf{s}}_3, \equiv'\}$, respectively, as shown below:

$$
\begin{array}{rc}
 & [A' \parr A''] \triangleleft [B' \parr B''] \\ \cline{2-2}
\hat{\mathsf{q}}_3^{\mathsf{R}}{\downarrow} & \langle [A' \parr A''] \triangleleft B'\rangle \parr B'' \\ \cline{2-2}
\hat{\mathsf{q}}_3^{\mathsf{L}}{\downarrow} & \langle A' \triangleleft B'\rangle \parr A'' \parr B'' \\ \cline{2-2}
\hat{\mathsf{q}}_2{\downarrow} & A' \parr B' \parr A'' \parr B'' \\ \cline{2-2}
\equiv' & [A' \parr A''] \parr [B' \parr B'']
\end{array}
\qquad
\begin{array}{rc}
 & [A' \parr A''] \otimes [B' \parr B''] \\ \cline{2-2}
\equiv', \hat{\mathsf{s}}_3, \equiv' & ([A' \parr A''] \otimes B') \parr B'' \\ \cline{2-2}
\hat{\mathsf{s}}_3 & (A' \otimes B') \parr A'' \parr B'' \\ \cline{2-2}
\hat{\mathsf{s}}_2 & A' \parr B' \parr A'' \parr B'' \\ \cline{2-2}
\equiv' & [A' \parr A''] \parr [B' \parr B'']
\end{array}
\qquad
\begin{array}{rc}
 & [A \parr [C' \parr C'']] \otimes B \\ \cline{2-2}
\equiv' & [[A \parr C'] \parr C''] \otimes B \\ \cline{2-2}
\hat{\mathsf{s}}_3 & ([A \parr C'] \otimes B) \parr C'' \\ \cline{2-2}
\hat{\mathsf{s}}_3 & [(A \otimes B) \parr C'] \parr C'' \\ \cline{2-2}
\equiv' & (A \otimes B) \parr [C' \parr C'']
\end{array}
$$

and similarly, the rules $\mathsf{q}_3^{\mathsf{L}}{\downarrow}$ and $\mathsf{q}_3^{\mathsf{R}}{\downarrow}$ are derivable in $\{\hat{\mathsf{q}}_3^{\mathsf{L}}{\downarrow}, \equiv'\}$ and $\{\hat{\mathsf{q}}_3^{\mathsf{R}}{\downarrow}, \equiv'\}$, respectively. ◀

4 BV is Contained in Pomset Logic

In this section we do not only show that every theorem of BV is also a theorem of pomset logic, but also that every proof in BV uniquely determines a pomset logic proof net with the same conclusion.

We have already seen in Section 2 that every formula uniquely determines a dicograph. Furthermore, by inspecting the rules of BV in Figure 2, one can see that the rule $\equiv$ does not change that dicograph, and that the rules s and $\mathsf{q}{\downarrow}$ only change the set of edges but not the set of vertices of the corresponding dicograph. Additionally, every instance of $\mathsf{ai}{\downarrow}$ removes one pair of dual atoms, and in a proof of BV, every atom occurring in the conclusion has to be removed by exactly one instance of $\mathsf{ai}{\downarrow}$ in the proof.

This means that every BV proof δ uniquely determines an axiom linking $\ell(\delta)$ for its conclusion, and hence, by definition a pomset logic pre-proof and also a relational RB-prenet.

We are now going to show that every relational RB-prenet that is obtained from a BV proof in such a way is indeed correct, and therefore every theorem of BV is also a theorem of pomset logic. The proof of the main lemma is based on the construction from [28], but the complete proof has never been published.

To begin, let δ be a BV proof of a formula A. We denote by $\llparenthesis \delta \rrparenthesis = \rho(A, \ell(\delta))$ the relational RB-prenet generated from δ as described in Section 2. Then the main result of this section is the following.

▶ **Theorem 4.1.** *For every* BV *proof* δ*, the relational RB-prenet* $\llparenthesis \delta \rrparenthesis$ *is correct.*

▶ **Example 4.2.** The three correct relational RB-prenets in the middle of (4) are obtained from the three BV-proofs in Example 3.1.

In order to prove Theorem 4.1, we first introduce an additional definition.

▶ **Definition 4.3.** *A formula is* balanced *if every propositional variable that occurs in* A *occurs exactly once positive and exactly once negative. A balanced formula* A *uniquely determines an axiom linking on* A*, that we denote by* $\ell(A)$*. Then we write* $\llparenthesis A \rrparenthesis$ *for the relational RB-prenet* $\rho(A, \ell(A))$*, i.e.,* $\llparenthesis A \rrparenthesis = (V_{\mathcal{A}}, R_{\mathcal{A}}, B_{\mathcal{A}})$*, where* $(V_{\mathcal{A}}, R_{\mathcal{A}}) = \llbracket A \rrbracket$ *and* $B_{\mathcal{A}}$ *is the matching associated to* $\ell(A)$*.*

Conversely, every RB-dicograph uniquely determines a balanced formula, up to renaming of variables and equivalence under $\equiv$. This gives us immediately the following proposition.

▶ **Proposition 4.4.** *Let* δ *be a proof in* BV*. Then there is a balanced formula* A*, that is provable in* BV *and such that* $\llparenthesis A \rrparenthesis$ *and* $\llparenthesis \delta \rrparenthesis$ *are isomorphic.*

Proof. Let B be the conclusion of δ. Then A is obtained from B by renaming all variable occurrences such that the result is balanced and the linking is preserved. ◀

▶ **Definition 4.5.** *Let* A *be a formula. A formula* B *is a* ***pseudo-subformula*** *of* A*, written as* $B \sqsubseteq A$*, if it is equivalent under* $\equiv$ *to some* A' *that can be obtained from* A *by replacing some atom occurrences in* A *by* $\mathbb{I}$*. If* $B \sqsubseteq A$ *and* $B \not\equiv A$*, then we say that* B *is a* ***proper pseudo-subformula*** *of* A*, and write it as* $B \sqsubset A$*.*

▶ **Example 4.6.** We have that $\langle (a \otimes b) \lhd d \lhd e \rangle \parr (b \otimes [(e \otimes f) \parr \langle a \lhd b \rangle])$ has $\langle a \lhd d \rangle \parr (b \otimes b)$ as pseudo-subformula which is equivalent to $\langle (a \otimes \mathbb{I}) \lhd d \lhd \mathbb{I} \rangle \parr (b \otimes [(\mathbb{I} \otimes \mathbb{I}) \parr \langle \mathbb{I} \lhd b \rangle])$.

The following proposition explains our choice to denote both pseudo-subformulas and induced subgraphs (Definition 2.5) by $\sqsubseteq$.

▶ **Proposition 4.7.** *We have* $B \sqsubseteq A$ *iff* $\llbracket B \rrbracket \sqsubseteq \llbracket A \rrbracket$ *and* $B \sqsubset A$ *iff* $\llbracket B \rrbracket \sqsubset \llbracket A \rrbracket$*.*

Proof. This follows directly from the definitions of $\llbracket \cdot \rrbracket$ and $\sqsubseteq$ and Proposition 2.4. ◀

▶ **Lemma 4.8.** *Let* A *be a balanced formula and* B *be a balanced pseudo-subformula of* A*. If* A *is provable in* BV*, then so is* B*.*

Proof. Let δ be the proof of A in BV, and let δ' be obtained by replacing all atoms that do not occur in B in every line of δ by $\mathbb{I}$. Then δ' is a valid derivation of B in BV. ◀

▶ **Definition 4.9.** *A* ***balanced cycle*** *is a balanced formula* H *such that* $\llparenthesis H \rrparenthesis$ *is an æ-cycle.*

▶ **Proposition 4.10.** *A formula* H *is a balanced cycle if and only if there are pairwise distinct atoms* $a_1, \ldots, a_n$ *for some* $n \geq 1$*, such that* $H \equiv L_1 \parr L_2 \parr \cdots \parr L_n$*, where* $L_1 = a_n^\perp \otimes a_1$ *or* $L_1 = a_n^\perp \lhd a_1$*, and for every* $i \in \{2, \ldots, n\}$ *we have* $L_i = a_{i-1}^\perp \otimes a_i$ *or* $L_i = a_{i-1}^\perp \lhd a_i$*.*

Proof. This follows immediately from the definitions. ◀

▶ **Definition 4.11.** *We say that a balanced formula A* ***contains a cycle*** *if it has a pseudo-subformula $B \sqsubseteq A$ that is a balanced cycle (or, equivalently if $(\!\![A]\!\!)$ contains a chordless æ-cycle).*

We are now ready to state and prove the central lemma to this section.

▶ **Lemma 4.12.** *Let $\mathsf{r}\,\dfrac{Q}{P}$ be an instance of an inference rule in* $\mathsf{BV\hat{u}}$*. If P is a balanced cycle then Q contains a cycle. If $\mathsf{r} \neq\ \equiv'$ then the size of the cycle in Q is strictly smaller than $|P|$.*

Proof. By Proposition 4.10 we have that $P \equiv' L_1 \parr L_2 \parr \ldots \parr L_n$, where $L_1 = a_n^\perp \otimes a_1$ or $L_1 = a_n^\perp \triangleleft a_1$, and for every $i \in \{2, \ldots, n\}$ we have $L_i = a_{i-1}^\perp \otimes a_i$ or $L_i = a_{i-1}^\perp \triangleleft a_i$, with all a_i being pairwise distinct. We proceed by case analysis on the rule r. First observe that by Proposition 4.10 the rules $\mathsf{ai}^{\otimes}{\downarrow}$, $\mathsf{ai}^{\triangleleft}_{\mathsf{L}}{\downarrow}$, $\mathsf{ai}^{\triangleleft}_{\mathsf{R}}{\downarrow}$ cannot be applied to P (seen bottom up), and if $\mathsf{r} = \equiv'$, then Q trivially contains a cycle, whose size is equal to $|P|$. Now assume r is

- $\mathsf{q}_4{\downarrow}\,\dfrac{[A \parr C] \triangleleft [B \parr D]}{\langle A \triangleleft B\rangle \parr \langle C \triangleleft D\rangle}$: Without loss of generality, assume that $A = a_n^\perp$ and $B = a_1$ and $C = a_{i-1}^\perp$ and $D = a_i$ for some $i \in \{2, \ldots, n\}$. Then

  $$Q \equiv' \langle [a_n^\perp \parr a_{i-1}^\perp] \triangleleft [a_1 \parr a_i]\rangle \parr L_2 \parr \cdots \parr L_{i-1} \parr L_{i+1} \parr \cdots \parr L_n$$

  which contains the cycle $\langle a_n^\perp \triangleleft a_i\rangle \parr L_{i+1} \parr \cdots \parr L_n$.
- $\hat{\mathsf{q}}_3^{\mathsf{L}}{\downarrow}\,\dfrac{[A \parr C] \triangleleft B}{\langle A \triangleleft B\rangle \parr C}$: Without loss of generality, we assume that $A = a_n^\perp$ and $B = a_1$ and $C = L_i$ for some $i \in \{2, \ldots, n\}$. Then

  $$Q \equiv' \langle [a_n^\perp \parr L_i] \triangleleft a_1\rangle \parr L_2 \parr \cdots \parr L_{i-1} \parr L_{i+1} \parr \cdots \parr L_n$$

  which contains the cycle $\langle a_{i-1}^\perp \triangleleft a_1\rangle \parr L_2 \parr \cdots \parr L_{i-1}$.
- $\hat{\mathsf{q}}_3^{\mathsf{R}}{\downarrow}\,\dfrac{A \triangleleft [B \parr C]}{\langle A \triangleleft B\rangle \parr C}$: As before, without loss of generality, we assume that $A = a_n^\perp$ and $B = a_1$ and $C = L_i$ for some $i \in \{2, \ldots, n\}$. Then

  $$Q \equiv' \langle a_n^\perp \triangleleft [a_1 \parr L_i]\rangle \parr L_2 \parr \cdots \parr L_{i-1} \parr L_{i+1} \parr \cdots \parr L_n$$

  which contains the cycle $\langle a_n^\perp \triangleleft a_i\rangle \parr L_{i+1} \parr \cdots \parr L_n$.
- $\hat{\mathsf{q}}_2{\downarrow}\,\dfrac{A \triangleleft B}{A \parr B}$: We can assume that $A = L_i$ and $B = L_j$ for some $i, j \in \{1, \ldots, n\}$. There are two subcases:
  - $i < j$: Then $Q = \langle L_i \triangleleft L_j\rangle \parr L_1 \parr \cdots \parr L_{i-1} \parr L_{i+1} \parr \cdots \parr L_{j-1} \parr L_{j+1} \parr \ldots \parr L_n$ which contains the cycle $L_1 \parr \cdots \parr L_{i-1} \parr \langle a_{i-1}^\perp \triangleleft a_j\rangle \parr L_{j+1} \parr \ldots \parr L_n$.
  - $j < i$: Then $Q = \langle L_i \triangleleft L_j\rangle \parr L_1 \parr \cdots \parr L_{j-1} \parr L_{j+1} \parr \cdots \parr L_{i-1} \parr L_{i+1} \parr \ldots \parr L_n$ which contains the cycle $\langle a_{i-1}^\perp \triangleleft a_j\rangle \parr L_{j+1} \parr \ldots \parr L_{i-1}$.
- $\hat{\mathsf{s}}_3\,\dfrac{[A \parr C] \otimes B}{(A \otimes B) \parr C}$: This case is analogous to the case $\hat{\mathsf{q}}_3^{\mathsf{L}}{\downarrow}$ above.
- $\hat{\mathsf{s}}_2\,\dfrac{A \otimes B}{A \parr B}$: This case is analogous to the case $\hat{\mathsf{q}}_2{\downarrow}$ above.

In all cases the size of the cycle in Q is strictly smaller than $|Q| = |P|$. ◀

▶ **Lemma 4.13.** *Let P be a balanced formula that contains a cycle. Then P is not provable in* BV*.*

Proof. Let H be the cycle in P, and let $n = |H|$ be its size. We proceed by induction on n. Note that n has to be even. For $n = 2$, we have that $H \equiv a^\perp \triangleleft a$ or $H \equiv a^\perp \otimes a$ for some atom a. By way of contradiction, assume P is provable in BV. By Lemma 4.8, H is also provable in BV, which is impossible. For the inductive case let now $n > 2$. As before, we have by Lemma 4.8 that H is provable in BV. By Proposition 3.3 and Proposition 3.5, H is provable in BVû. Let δ be that proof in BVû. Let now Q be the premise of the bottommost rule instance r of δ that is not a $\equiv'$ (i.e., the conclusion of r is $H' \equiv' H$ and $Q \not\equiv' H$). By Lemma 4.12, Q contains a cycle whose size is smaller than n. By induction hypothesis Q is not provable in BV, and therefore also not provable in BVû, which is a contradiction to the existence to δ. ◀

We can now complete the proof of Theorem 4.1.

Proof of Theorem 4.1. Let δ be a proof in BV. By Proposition 4.4, there is a balanced formula P, such that $\llbracket P \rrbracket$ is isomorphic to $\llbracket \delta \rrbracket$, and such that P is provable in BV. Now assume, by way of contradiction, that $\llbracket \delta \rrbracket$ is incorrect. That means that $\llbracket \delta \rrbracket$ contains a chordless æ-cycle, or equivalently, that P contains a cycle. By Lemma 4.13, P is not provable in BV. Contradiction. ◀

5 Pomset Logic is not Contained in BV

In this section we present a formula that is provable in pomset logic, i.e., has a correct pomset logic proof net, but that is not provable in BV. From what has been said in the previous section, it follows that if such a formula exists then there is also a balanced such formula. The formula we discuss in this section is the formula Q shown below:

$$Q = (\langle a \triangleleft b\rangle \otimes \langle c \triangleleft d\rangle) \mathbin{\bindnasrepma} (\langle e \triangleleft f\rangle \otimes \langle g \triangleleft h\rangle) \mathbin{\bindnasrepma} \langle a^\perp \triangleleft h^\perp\rangle \mathbin{\bindnasrepma} \langle e^\perp \triangleleft b^\perp\rangle \mathbin{\bindnasrepma} \langle g^\perp \triangleleft d^\perp\rangle \mathbin{\bindnasrepma} \langle c^\perp \triangleleft f^\perp\rangle \quad (10)$$

or equivalently, the sequent

$$\Gamma_Q = [\langle a \triangleleft b\rangle \otimes \langle c \triangleleft d\rangle, \langle e \triangleleft f\rangle \otimes \langle g \triangleleft h\rangle, a^\perp \triangleleft h^\perp, e^\perp \triangleleft b^\perp, g^\perp \triangleleft d^\perp, c^\perp \triangleleft f^\perp] \quad (11)$$

Since the formula Q (resp. the sequent Γ_Q) is balanced, there is a unique axiom linking and therefore a unique relational RB-prenet and a unique tree-like RB-prenet. In Figure 4, we show the tree-like RB prenet for Γ_Q, and on the left of Figure 5 we show the relational RB-prenet, which is the same for Q and Γ_Q.

To see that these are provable in pomset logic, we have to show that the RB-prenets do not contain chordless æ-cycles. For this we focus on the tree-like RB-prenet, because in tree-like RB-prenets all æ-paths (and therefore also all æ-cycles) are chordless. Hence, it suffices to show that there are no æ-cycles.

Observe that the B-edges corresponding to the roots of the formulas in Γ_Q cannot participate in an æ-cycle because they have no adjacent R-edge at the bottom. We can therefore remove each of these B-edges, together with the two adjacent R-edges at the top. The resulting graph is shown on the right of Figure 5.

Another simplification we can do without affecting the æ-cycles in the graph is replacing the two B-edges labeled $a \triangleleft b$ and $c \triangleleft d$, together with the connecting R-edge by a single B-edge, and similarly for the two B-edges $g \triangleleft h$ and $e \triangleleft f$. The result is shown on the left of Figure 6.

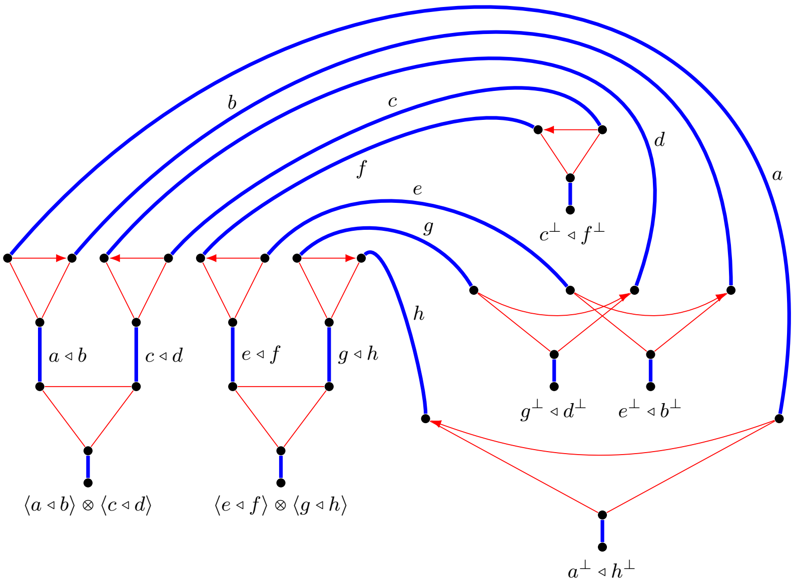


Figure 4 The tree-like RB-preenet for the sequent Γ_Q in Equation (11).

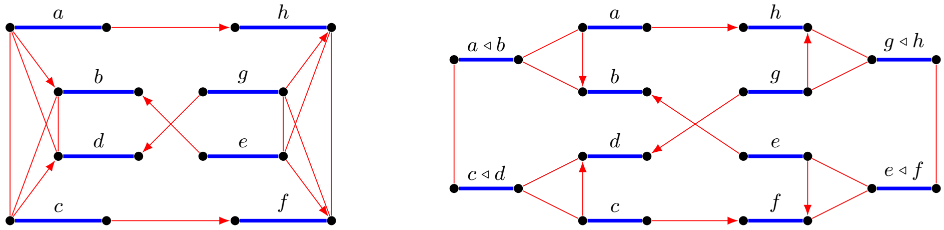


Figure 5 Left: The relational RB-prenet for Q in (10) and Γ_Q in (11).
Right: A simplification of the tree-like RB-prenet in Figure 4.

Finally, observe that there is no æ-cycle that passes trough the two B-edges labeled b and a. The reason is that the directed R-edge between them has the opposite direction of the two adjacent R-edges on the other endpoints of these B-edges. Thus, we can collapse these two edges (and the adjacent "triangle") to a single vertex. The same can be done for the pairs c/d and g/h and e/f. The result of this operation is shown on the right of Figure 6.

▶ **Proposition 5.1.** *The formula Q and the sequent Γ_Q shown in Equation* (10) *and Equation* (11) *above are provable in pomset logic.*

Proof. In the paragraphs above, we have argued that the tree-like RB-prenet in Figure 4 has an æ-cycle if and only if the RB-digraph on the right of Figure 6 has an æ-cycle. Now it is easy to see that this graph has no æ-cycle. Hence, tree-like RB-prenet for Γ_Q is correct. ◀

Let us now show that the formula Q is not provable in BV. To do so we will show that whenever a BV inference has as conclusion Q then its premise defines an incorrect RB-prenet in pomset logic, and is therefore not provable in pomset logic. Since by Theorem 4.1 all BV proofs induce correct pomset proof nets, we can conclude that those premises are not BV-provable, therefore there is no way to build a BV-proof of Q.

The main difficulty here is to make sure that we do not overlook any case when checking all possible inferences that have Q as conclusion. Since the unit $\mathbb{I}$ can make these kind of arguments difficult to check, we use here BVû. Now observe that Q has no subformula of the form $x \parr x^{\perp}$. This means we only have to consider the non-axiom rules of BVû.

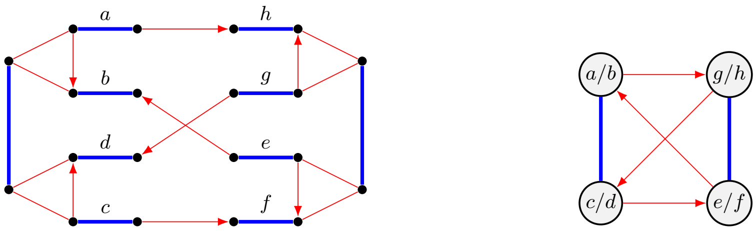


Figure 6 Two further simplifications of the graph on the right of Figure 5.

To cut down the number of cases to consider, we take advantage of the symmetries of Q. Let us first look at the *automorphisms*, i.e., permutations of the variables that results in a formula Q' with $Q' \equiv Q$, which means $\llbracket Q' \rrbracket = \llbracket Q \rrbracket$. The following are automorphisms:

(α) $a \leftrightarrow c,\ b \leftrightarrow d,\ e \leftrightarrow g,\ f \leftrightarrow h$

(β) $a \mapsto e,\ b \mapsto f,\ c \mapsto g,\ d \mapsto h,\ e \mapsto c,\ f \mapsto d,\ g \mapsto a,\ h \mapsto b$

The action of these automorphisms on the subformulas of Q of the form $x^\perp \lhd y^\perp$ is transitive: $\alpha(a^\perp \lhd h^\perp) = c^\perp \lhd f^\perp$, $\beta(a^\perp \lhd h^\perp) = e^\perp \lhd b^\perp$ and $\alpha \circ \beta(a^\perp \lhd h^\perp) = g^\perp \lhd d^\perp$.

Another useful symmetry is not quite an automorphism: it is the following *anti-automorphism*:

(γ) $a \leftrightarrow h,\ b \leftrightarrow g,\ c \leftrightarrow f,\ d \leftrightarrow e$

that sends Q to its “conjugate” $Q^\dagger$ defined inductively as follows:

$$x^\dagger = x \text{ when } x \text{ is an atom} \qquad (B \odot C)^\dagger = C^\dagger \odot B^\dagger \text{ for } \odot \in \{\parr, \otimes, \lhd\}$$

Note that the reversal of the arguments only matters for the non-commutative connective $\lhd$, and $\llbracket Q^\dagger \rrbracket$ is the same as $\llbracket Q \rrbracket$, except that all directed R-edges have the opposite direction. Thus, conjugacy preserves provability both in pomset logic (reversing the direction of all cycles in the correctness criterion) and in system $\mathsf{BV\hat{u}}$ (the inference rules are closed under conjugacy, with $\hat{\mathsf{q}}^{\mathsf{L}}_3\downarrow$ and $\hat{\mathsf{q}}^{\mathsf{R}}_3\downarrow$ being swapped).

We will now go through all the rules of $\mathsf{BV\hat{u}}$ and check all possible applications. Using a similar argument as in the proof of Lemma 4.12, we will see that in each case there is a cycle in the resulting premise.

- $\mathsf{q}_4\downarrow \dfrac{[A \parr C] \lhd [B \parr D]}{\langle A \lhd B\rangle \parr \langle C \lhd D\rangle}$: Because of the action of the automorphisms α/β, we can without loss of generality assume that $A = a^\perp$ and $B = h^\perp$. There are three subcases:
  - $C = e^\perp$ and $D = b^\perp$. We get the cycle $(e \otimes h) \parr \langle e^\perp \lhd h^\perp\rangle$ in the premise of the $\mathsf{q}_4\downarrow$-application.
  - $C = g^\perp$ and $D = d^\perp$. We get the cycle $(a \otimes d) \parr \langle a^\perp \lhd d^\perp\rangle$ in the premise of the $\mathsf{q}_4\downarrow$-application.
  - $C = c^\perp$ and $D = f^\perp$. We get the cycle $(b \otimes c) \parr (e \otimes h) \parr \langle c^\perp \lhd h^\perp\rangle \parr \langle e^\perp \lhd b^\perp\rangle$ in the premise of the $\mathsf{q}_4\downarrow$-application.
- $\hat{\mathsf{q}}^{\mathsf{L}}_3\downarrow \dfrac{[A \parr C] \lhd B}{\langle A \lhd B\rangle \parr C}$: As before, because of the symmetries of Q, we only need to consider the case where $A = a^\perp$ and $B = h^\perp$. There are now five subcases of how to match C:
  - $C = \langle a \lhd b\rangle \otimes \langle c \lhd d\rangle$. We get the cycle $(e \otimes h) \parr \langle b \lhd h^\perp\rangle \parr \langle e^\perp \lhd b^\perp\rangle$ in the premise of the $\hat{\mathsf{q}}^{\mathsf{L}}_3\downarrow$-application.
  - $C = \langle e \lhd f\rangle \otimes \langle g \lhd h\rangle$. We get the cycle $h \lhd h^\perp$ in the premise of the $\hat{\mathsf{q}}^{\mathsf{L}}_3\downarrow$-application.
  - $C = e^\perp \lhd b^\perp$. We get the cycle $(e \otimes h) \parr \langle e^\perp \lhd h^\perp\rangle$ in the premise of the $\hat{\mathsf{q}}^{\mathsf{L}}_3\downarrow$-application.

  - $C = g^{\perp} \lhd d^{\perp}$. We get the cycle $(b \otimes d) \parr (e \otimes h) \parr \langle d^{\perp} \lhd h^{\perp} \rangle \parr \langle e^{\perp} \lhd b^{\perp} \rangle$ in the premise of the $\hat{\mathsf{q}}_3^{\mathsf{L}}\downarrow$-application.
  - $C = c^{\perp} \lhd f^{\perp}$. We get the cycle $(f \otimes h) \parr \langle f^{\perp} \lhd h^{\perp} \rangle$ in the premise of the $\hat{\mathsf{q}}_3^{\mathsf{L}}\downarrow$-application.
- $\hat{\mathsf{q}}_3^{\mathsf{R}}\downarrow \dfrac{A \lhd [B \parr C]}{\langle A \lhd B \rangle \parr C}$: Similar to $\hat{\mathsf{q}}_3^{\mathsf{L}}\downarrow$, by conjugacy.
- $\hat{\mathsf{q}}_2\downarrow \dfrac{A \lhd B}{A \parr B}$: The possible values for the ordered pair (A, B) are all pairs of distinct formulas in the sequent Γ_Q in Equation (11). We first look at the case $A = \langle a \lhd b \rangle \otimes \langle c \lhd d \rangle$ and $B = \langle e \lhd f \rangle \otimes \langle g \lhd h \rangle$. Here we get the cycle $\langle d \lhd g \rangle \parr \langle g^{\perp} \lhd d^{\perp} \rangle$ in the premise. The case $A = \langle e \lhd f \rangle \otimes \langle g \lhd h \rangle$ and $B = \langle a \lhd b \rangle \otimes \langle c \lhd d \rangle$ is symmetric to the this one via the automorphism β. Otherwise, either A or B (or both) have the form $x^{\perp} \lhd y^{\perp}$. It suffices to treat all the cases $R = x^{\perp} \lhd y^{\perp}$. This is because conjugation exchanges the roles of A and B in the $\mathsf{q}_2\downarrow$-rule, and Q is equal to its own conjugate up to the variable renaming performed by γ. We may also without loss of generality assume that $A = a^{\perp} \lhd h^{\perp}$; as before, this relies on the transitive action of the automorphisms of Q on the $x^{\perp} \lhd y^{\perp}$ that it contains. There are now five cases for B:
  - $B = \langle a \lhd b \rangle \otimes \langle c \lhd d \rangle$. We get the cycle $a^{\perp} \lhd a$ in the premise.
  - $B = \langle e \lhd f \rangle \otimes \langle g \lhd h \rangle$. We get the cycle $h^{\perp} \lhd h$ in the premise.
  - $B = e^{\perp} \lhd b^{\perp}$. We get the cycle $(e \otimes h) \parr \langle h^{\perp} \lhd e^{\perp} \rangle$ in the premise.
  - $B = g^{\perp} \lhd d^{\perp}$. We get the cycle $(a \otimes d) \parr \langle a^{\perp} \lhd d^{\perp} \rangle$ in the premise.
  - $B = c^{\perp} \lhd f^{\perp}$. We get the cycle $(f \otimes h) \parr \langle h^{\perp} \lhd f^{\perp} \rangle$ in the premise.
- $\hat{\mathsf{s}}_3 \dfrac{[A \parr C] \otimes B}{(A \otimes B) \parr C}$: There are two possibilities to match $A \otimes B$: either with $\langle a \lhd b \rangle \otimes \langle c \lhd d \rangle$ or with $\langle e \lhd f \rangle \otimes \langle g \lhd h \rangle$. Due to the commutativity of $\otimes$, we have four possibilities to match A and B. Due to the symmetries discussed above, we only need to consider the case where $A = a \lhd b$ and $B = c \lhd d$. There are now five cases how to match C:
  - $C = \langle e \lhd f \rangle \otimes \langle g \lhd h \rangle$. We get the cycle $(f \otimes c) \parr \langle c^{\perp} \lhd f^{\perp} \rangle$ in the premise.
  - $C = a^{\perp} \lhd h^{\perp}$. We get the cycle $(h^{\perp} \otimes c) \parr \langle c^{\perp} \lhd f^{\perp} \rangle \parr (f \otimes h)$ in the premise.
  - $C = e^{\perp} \lhd b^{\perp}$. We get the cycle $(e^{\perp} \otimes d) \parr \langle g^{\perp} \lhd d^{\perp} \rangle \parr (e \otimes g)$ in the premise.
  - $C = g^{\perp} \lhd d^{\perp}$. We get the cycle $d^{\perp} \otimes d$ in the premise.
  - $C = c^{\perp} \lhd f^{\perp}$. We get the cycle $c^{\perp} \otimes c$ in the premise.
- $\hat{\mathsf{s}}_2 \dfrac{A \otimes B}{A \parr B}$: This case is already subsumed by the case for $\hat{\mathsf{q}}_2\downarrow$.

In this way, we have completed the proof of the following proposition.

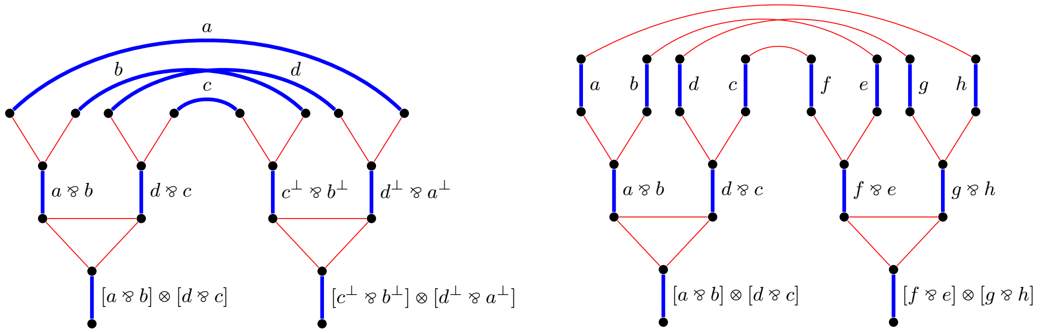
▶ **Proposition 5.2.** *The formula Q shown in Equation* (10) *is not provable in* BV.

▶ **Theorem 5.3.** *The theorems of* BV *form a proper subset of the theorems of pomset logic.*

Proof. This follows immediately from Propositions 5.1 and 5.2. ◀

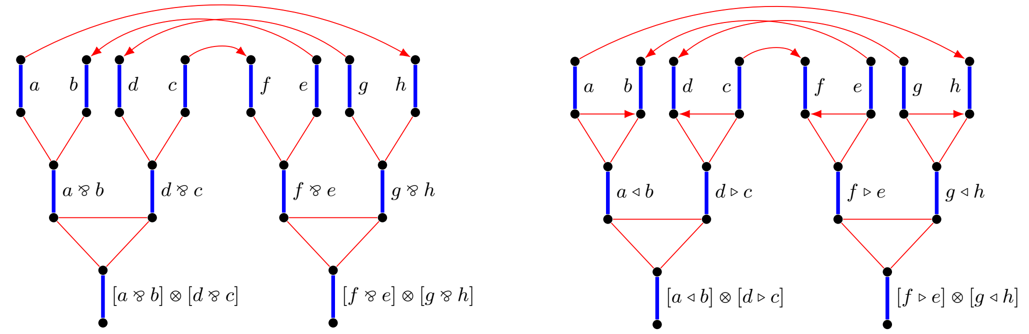
6 Conclusion

Let us end this paper by giving some historical perspective and some explanation how the formula Q has been found. The main reason that it took more than 20 year to find this (rather simple) formula was that everyone (including the second author) was looking into the wrong direction, trying to prove that BV and pomset logic are the same. This changed only after the first author (not being aware of the pomset logic vs. BV problem) observed



(a) A tree-like RB-prenet for a linear version of the medial rule of system SKS. Note that it does *not* satisfy the MLL_0 correctness criterion, and therefore also not the pomset criterion.

(b) A variation of the prenet on the left. The undirected R-edges on the top correspond to the addition of $a^\perp \otimes h^\perp, b^\perp \otimes e^\perp, d^\perp \otimes g^\perp, c^\perp \otimes f^\perp$. Note that the prenet is still *not* correct.



(c) The R-edges on top are now directed, corresponding to $a^\perp \lhd h^\perp, b^\perp \rhd e^\perp, d^\perp \rhd g^\perp, c^\perp \lhd f^\perp$. This modification validates the pomset logic correctness criterion, but the resulting sequent is not provable in BV.

(d) Adding more R-edges does preserve provability in pomset logic, but showing that the resulting sequent is not provable in BV is easier now, as every possible rule application breaks pomset correctness.

Figure 7 From the medial of SKS to our counterexample.

that checking pomset logic correctness is **coNP**-complete [18]. Since it had been observed before that BV is **NP**-complete [15], this immediately entailed that either **NP** = **coNP** or BV $\neq$ pomset logic.

Unfolding and dissecting the proof of **coNP**-completeness of pomset logic correctness led to a relation to classical logic provability. The details of this are subject of ongoing research and would go beyond the scope of this paper. But the outcome let us to the study of linear inferences [5, 6] which are a special case of balanced tautologies [27]. We were looking at linear inferences that are tautologies in classical logic but not provable in linear logic. The simplest such inference is $(A \wedge D) \vee (B \wedge C) \Rightarrow [A \vee B] \wedge [D \vee C]$, which corresponds to the medial rule of system SKS [2], a formulation of classical logic in the calculus of structures. Its linear version $(A \otimes D) \parr (B \otimes C) \multimap [A \parr B] \otimes [D \parr C]$ is, of course, not a theorem of MLL_0. This can be immediately seen by inspecting the RB-prenet for the formula $(a \otimes d) \parr (b \otimes c) \multimap [a \parr b] \otimes [c \parr d]$, which is shown in Figure 7a, and which contains several (chordless) æ-cycles. Then, on the right of that "medial RB-prenet", in Figure 7b, we replace the B-edges corresponding to the atoms by a pair of B-edges connected by an (undirected) R-edge. This does not affect provability, as no æ-cycles are added or removed. Then, in Figure 7c, we give these new R-edges a direction. By choosing the "right" direction, we can break all æ-cycles, which means the result becomes correct with respect to the pomset logic correctness criterion. But the resulting formula (or sequent) remains unprovable in

BV. To simplify the proof of non-provability in BV, we added further R-edges, as shown in Figure 7d, that do not break provability in pomset logic. The result is an intermediate step between the RB-prenets in Figure 4 and Figure 5.

The knowledge that BV and pomset logic are different, leads to four immediate open problems: (i) can we find a proof net correctness criterion for BV, (ii) can we find a deductive proof system for pomset logic that is independent from the prenets[9], (iii) which of the two logics is better, and (iv) are these two the only ones, or are there more logics having these three connectives and being conservative over MLL_0?

References

1 V. Michele Abrusci and Paul Ruet. Non-commutative logic I: the multiplicative fragment. *Annals of Pure and Applied Logic*, 101(1):29–64, 1999. `doi:10.1016/S0168-0072(99)00014-7`.

2 Kai Brünnler and Alwen Fernanto Tiu. A local system for classical logic. In Robert Nieuwenhuis and Andrei Voronkov, editors, *Logic for Programming, Artificial Intelligence, and Reasoning, 8th International Conference, LPAR 2001, Havana, Cuba, December 3-7, 2001, Proceedings*, volume 2250 of *Lecture Notes in Computer Science*, pages 347–361. Springer, 2001. `doi:10.1007/3-540-45653-8_24`.

3 Paola Bruscoli. A purely logical account of sequentiality in proof search. In Peter J. Stuckey, editor, *Logic Programming, 18th International Conference, ICLP 2002, Copenhagen, Denmark, July 29 - August 1, 2002, Proceedings*, volume 2401 of *Lecture Notes in Computer Science*, pages 302–316. Springer, 2002. `doi:10.1007/3-540-45619-8_21`.

4 Denis Béchet, Philippe de Groote, and Christian Retoré. A complete axiomatisation for the inclusion of series-parallel partial orders. In Hubert Comon, editor, *Rewriting Techniques and Applications, 8th International Conference, RTA-97, Sitges, Spain, June 2-5, 1997, Proceedings*, volume 1232 of *Lecture Notes in Computer Science*, pages 230–240. Springer, 1997. `doi:10.1007/3-540-62950-5_74`.

5 Anupam Das and Alex A. Rice. New minimal linear inferences in boolean logic independent of switch and medial. In Naoki Kobayashi, editor, *6th International Conference on Formal Structures for Computation and Deduction, FSCD 2021, July 17-24, 2021, Buenos Aires, Argentina (Virtual Conference)*, volume 195 of *LIPIcs*, pages 14:1–14:19. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2021. `doi:10.4230/LIPIcs.FSCD.2021.14`.

6 Anupam Das and Lutz Straßburger. On linear rewriting systems for boolean logic and some applications to proof theory. *Logical Methods in Computer Science*, 12(4), 2016. `doi:10.2168/LMCS-12(4:9)2016`.

7 Arnaud Fleury and Christian Retoré. The mix rule. *Mathematical Structures in Computer Science*, 4(2):273–285, 1994. `doi:10.1017/S0960129500000451`.

8 Jean-Yves Girard. Linear logic. *Theoretical Computer Science*, 50(1):1–101, January 1987. `doi:10.1016/0304-3975(87)90045-4`.

9 Jean-Yves Girard. Towards a geometry of interaction. In J. W. Gray and A. Scedrov, editors, *Categories in Computer Science and Logic*, volume 92 of *Contemporary Mathematics*, pages 69–108. American Mathematical Society, Providence, RI, 1989. Proceedings of a Summer Research Conference held June 14–20, 1987. `doi:10.1090/conm/092/1003197`.

10 Alessio Guglielmi. A system of interaction and structure. *ACM Transactions on Computational Logic*, 8(1), January 2007. According to the author, the published version contains errors introduced by the editorial processing; the authoritative version is `arXiv:cs/9910023`. `doi:10.1145/1182613.1182614`.

11 Alessio Guglielmi and Lutz Straßburger. Non-commutativity and MELL in the Calculus of Structures. In Gerhard Goos, Juris Hartmanis, Jan van Leeuwen, and Laurent Fribourg, editors, *Computer Science Logic*, volume 2142, pages 54–68. Springer Berlin Heidelberg, Berlin, Heidelberg, 2001. `doi:10.1007/3-540-44802-0_5`.

---

[9] The sequent system proposed by Slavnov [25], uses labels for encoding the paths in the proof net.

12 Ross Horne and Alwen Tiu. Constructing weak simulations from linear implications for processes with private names. *Mathematical Structures in Computer Science*, 29(8):1275–1308, 2019. `doi:10.1017/S0960129518000452`.

13 Ross Horne, Alwen Tiu, Bogdan Aman, and Gabriel Ciobanu. De morgan dual nominal quantifiers modelling private names in non-commutative logic. *ACM Transactions on Computational Logic*, 20(4):22:1–22:44, 2019. `doi:10.1145/3325821`.

14 Ozan Kahramanoğulları. System BV without the Equalities for Unit. In *Computer and Information Sciences - ISCIS 2004*, volume 3280, pages 986–995. Springer Berlin Heidelberg, Berlin, Heidelberg, 2004. `doi:10.1007/978-3-540-30182-0_99`.

15 Ozan Kahramanoğulları. System BV is NP-complete. *Annals of Pure and Applied Logic*, 152(1):107–121, March 2008. `doi:10.1016/j.apal.2007.11.005`.

16 Joachim Lambek. The mathematics of sentence structure. *The American Mathematical Monthly*, 65(3):154–170, 1958. URL: `http://www.jstor.org/stable/2310058`.

17 Alain Lecomte and Christian Retoré. Pomset logic as an alternative categorial grammar. In *Formal Grammar*, pages 181–196. FoLLI, 1995.

18 Lê Thành Dũng Nguyên. Complexity of correctness for pomset logic proof nets, 2020. `arXiv:1912.10606`.

19 Peter W. O'Hearn and David J. Pym. The logic of bunched implications. *Bulletin of Symbolic Logic*, 5(2):215–244, 1999. `doi:10.2307/421090`.

20 Christian Retoré. *Réseaux et séquents ordonnés*. PhD thesis, Université Paris VII, February 1993. URL: `https://tel.archives-ouvertes.fr/tel-00585634`.

21 Christian Retoré. Pomset logic: A non-commutative extension of classical linear logic. In *Typed Lambda Calculi and Applications*, volume 1210, pages 300–318. Springer Berlin Heidelberg, Berlin, Heidelberg, 1997. `doi:10.1007/3-540-62688-3_43`.

22 Christian Retoré. Pomset Logic as a Calculus of Directed Cographs. Research Report 3714, INRIA, June 1999. URL: `https://hal.inria.fr/inria-00072953`.

23 Christian Retoré. Handsome proof-nets: perfect matchings and cographs. *Theoretical Computer Science*, 294(3):473–488, February 2003. `doi:10.1016/S0304-3975(01)00175-X`.

24 Christian Retoré. Pomset Logic: The other approach to noncommutativity in logic. In Claudia Casadio and Philip J. Scott, editors, *Joachim Lambek: the interplay of mathematics, logic, and linguistics*, Outstanding Contributions to Logic, chapter 9, pages 299–346. Springer, 2021. `doi:10.1007/978-3-030-66545-6_9`.

25 Sergey Slavnov. On noncommutative extensions of linear logic. *Logical Methods in Computer Science*, Volume 15, Issue 3, September 2019. `doi:10.23638/LMCS-15(3:30)2019`.

26 Lutz Straßburger. System NEL is undecidable. *Electronic Notes in Theoretical Computer Science*, 84:166–177, 2003. `doi:10.1016/S1571-0661(04)80853-3`.

27 Lutz Straßburger. Extension without cut. *Annals of Pure and Applied Logic*, 163(12):1995–2007, 2012. `doi:10.1016/j.apal.2012.07.004`.

28 Lutz Straßburger. *Linear Logic and Noncommutativity in the Calculus of Structures*. PhD Thesis, Technische Universität Dresden, 2003. URL: `https://www.lix.polytechnique.fr/~lutz/papers/dissvonlutz.pdf`.

29 David N. Yetter. Quantales and (noncommutative) linear logic. *The Journal of Symbolic Logic*, 55(1):41–64, March 1990. `doi:10.2307/2274953`.