

The Calculus of Structures

Alessio Guglielmi
Technische Universität Dresden
<http://alessio.guglielmi.name/>

The Calculus of Structures is a proof-theoretical formalism motivated by computation. Its most direct analogue is Gentzen's Sequent Calculus. We know that systems in the Sequent Calculus can be interpreted computationally, in the proof-normalisation-as-computation and proof-search-as-computation paradigms. Many aspects of computation are not managed satisfactorily by the Sequent Calculus, especially some important for distributed computation, like locality, atomicity and several kinds of modularity. This holds for all logics expressed in the Sequent Calculus, including, e.g., Linear Logic.

The Calculus of Structures addresses these problems in a radically new, purely proof-theoretical foundation, based on the principles of deep inference and top-down symmetry. We express classical, linear, modal logics, and several variants thereof, and their deductive systems are as simple as those in the Sequent Calculus, but possess the desired computational properties. This new proof theory is richer than the old one, e.g., there are new interesting notions of normalisation.

Atomic Cut Elimination for Classical Logic

Kai Brünnler

Technische Universität Dresden, Fakultät Informatik
D-01062 Dresden, Germany
`kai.bruennler@inf.tu-dresden.de`

Abstract. System SKS is a set of rules for classical propositional logic presented in the calculus of structures. Like sequent systems and unlike natural deduction systems, it has an explicit cut rule, which is admissible. In contrast to sequent systems, the cut rule can easily be reduced to atomic form. This allows for a very simple cut elimination procedure based on plugging in parts of a proof, like normalisation in natural deduction and unlike cut elimination in the sequent calculus. It should thus be a good common starting point for investigations into both proof search as computation and proof normalisation as computation.

Keywords: sequent calculus, natural deduction, cut elimination, classical logic, atomic cut.

1 Introduction

The two well-known connections between proof theory and language design, *proof search as computation* and *proof normalisation as computation*, have mainly used different proof-theoretic formalisms. While designers of functional programming languages prefer natural deduction, because of the close correspondence between proof normalisation and reduction in related term calculi [4,8], designers of logic programming languages prefer the sequent calculus [7], because infinite choice and much of the unwanted non-determinism is limited to the cut rule, which can be eliminated.

System SKS [2] is a set of inference rules for classical propositional logic presented in a new formalism, the *calculus of structures* [5]. This system admits the good properties usually found in sequent systems: in particular, all rules that induce infinite choice in proof search are admissible. Thus, in principle, it is as suitable for proof search as systems in the sequent calculus. In this paper I will present a cut elimination procedure for SKS that is very similar to normalisation in natural deduction. It thus allows us to develop, at least for the case of classical logic, both the proof search and the proof normalisation paradigm of computation in the same formalism and starting from the same system of rules.

Cut elimination in the sequent calculus and normalisation in natural deduction, widely perceived as ‘morally the same’, differ quite a bit, technically. Compared to cut elimination, (weak) normalisation is simpler, involving neither

Units	
Associativity	$[f, \mathbf{R}] = [\mathbf{R}]$ $(t, \mathbf{R}) = (\mathbf{R})$ $[t, t] = t$ $(f, f) = f$
$[\mathbf{R}, [\mathbf{T}]] = [\mathbf{R}, \mathbf{T}]$ $(\mathbf{R}, (\mathbf{T})) = (\mathbf{R}, \mathbf{T})$	
Commutativity	Negation
$[\mathbf{R}, \mathbf{T}] = [\mathbf{T}, \mathbf{R}]$ $(\mathbf{R}, \mathbf{T}) = (\mathbf{T}, \mathbf{R})$	$\bar{t} = f$ $\bar{f} = t$ $\frac{[\mathbf{R}_1, \dots, \mathbf{R}_h]}{(\bar{\mathbf{R}}_1, \dots, \bar{\mathbf{R}}_h)}$ $\frac{(\mathbf{R}_1, \dots, \mathbf{R}_h)}{[\bar{\mathbf{R}}_1, \dots, \bar{\mathbf{R}}_h]}$ $\bar{\bar{R}} = R$
Singleton	
$[R] = R = (R)$	

Fig. 1. Equations on structures

After introducing basic notions of the calculus of structures, I show system SKS with atomic contraction, weakening, identity and, most significantly, atomic cut. Then, after establishing some lemmas, I present the cut elimination procedure.

2 The Calculus of Structures

Definition 1. Propositional variables p and their negations $\bar{p}$ are *atoms*, with the negation of $\bar{p}$ defined to be p . Atoms are denoted by $a, b, \dots$. The *structures* of the language KS are generated by

$$S ::= t \mid f \mid a \mid \underbrace{[S, \dots, S]}_{>0} \mid \underbrace{(S, \dots, S)}_{>0} \mid \bar{S} \quad ,$$

where t and f are the units *true* and *false*, $[S_1, \dots, S_h]$ is a *disjunction* and $(S_1, \dots, S_h)$ is a *conjunction*. $\bar{S}$ is the *negation* of the structure S . The units are not atoms. Structures are denoted by S, R, T, U and V . *Structure contexts*, denoted by $S\{ \}$, are structures with one occurrence of $\{ \}$, the *empty context* or *hole*, that does not appear in the scope of a negation. $S\{R\}$ denotes the structure obtained by filling the hole in $S\{ \}$ with R . We drop the curly braces when they are redundant: for example, $S[R, T]$ stands for $S\{[R, T]\}$. Structures are *equivalent* modulo the smallest congruence relation induced by the equations shown in Fig. 1, where $\mathbf{R}$ and $\mathbf{T}$ are finite, non-empty sequences of structures. In general we do not distinguish between equivalent structures.

Definition 2. An *inference rule* is a scheme of the kind $\rho \frac{S\{T\}}{S\{R\}}$, where ρ is the *name* of the rule, $S\{T\}$ is its *premise* and $S\{R\}$ is its *conclusion*. In an instance

of ρ , the structure taking the place of R is called *redex* and the structure taking the place of T is called *contractum*. A (formal) *system* $\mathcal{S}$ is a set of inference rules. To clarify the use of the equational theory where it is not obvious, I will use the rule $= \frac{T}{R}$ where R and T are equivalent structures.

Definition 3. The *dual* of a rule is obtained by exchanging premise and conclusion and replacing each connective by its De Morgan dual.

Definition 4. A *derivation* Δ in a certain formal system is a finite chain of instances of inference rules in the system:

$$\begin{array}{c} \pi' \frac{T}{V} \\ \pi \text{ ---} \\ \vdots \\ \rho' \frac{\text{---}}{U} \\ \rho \frac{\text{---}}{R} \end{array} .$$

A derivation can consist of just one structure. The topmost structure in a derivation is called the *premise* of the derivation, and the structure at the bottom is called its *conclusion*. A derivation Δ whose premise is T , whose conclusion is R ,

and whose inference rules are in $\mathcal{S}$ will be indicated with $\Delta \parallel \frac{T}{R}$. A *proof* Π in

the calculus of structures is a derivation whose premise is the unit true. It will be denoted by $\Pi \parallel \frac{\text{---}}{R}$. A rule ρ is *derivable* for a system $\mathcal{S}$ if for every instance

of $\rho \frac{T}{R}$ there is a derivation $\parallel \frac{T}{R}$. A rule ρ is *admissible* for a system $\mathcal{S}$ if for

every proof $\parallel \frac{\text{---}}{S}^{\mathcal{S} \cup \{\rho\}}$ there is a proof $\parallel \frac{\text{---}}{S}^{\mathcal{S}}$.

3 System SKS

System SKS, shown in Fig. 2, has been introduced and shown to be sound and complete for classical propositional logic in [2]. The first S stands for “symmetric” or “self-dual”, meaning that for each rule, its dual (or contrapositive) is also in the system. The K stands for “klassisch” as in Gentzen’s LK and the last S says that it is a system on structures.

The rules $\text{ai}\downarrow, \text{s}, \text{m}, \text{aw}\downarrow, \text{ac}\downarrow$ are called respectively *atomic identity*, *switch*, *medial*, *atomic weakening* and *atomic contraction*. Their dual rules carry the same name prefixed with a “co-”, so e.g. $\text{aw}\uparrow$ is called *atomic co-weakening*. The rules s and m are their own duals. The rule $\text{ai}\uparrow$ is special, it is called *atomic*

$$\begin{array}{ccc}
\text{ai}\downarrow \frac{S\{t\}}{S[a, \bar{a}]} & & \text{ai}\uparrow \frac{S(a, \bar{a})}{S\{f\}} \\
\\
& \text{s} \frac{S([R, U], T)}{S[(R, T), U]} & \\
& \text{m} \frac{S([R, U], (T, V))}{S([R, T], [U, V])} & \\
\\
\text{aw}\downarrow \frac{S\{f\}}{S\{a\}} & & \text{aw}\uparrow \frac{S\{a\}}{S\{t\}} \\
\\
\text{ac}\downarrow \frac{S[a, a]}{S\{a\}} & & \text{ac}\uparrow \frac{S\{a\}}{S(a, a)}
\end{array}$$

Fig. 2. System SKS

$$\begin{array}{ccc}
\text{ai}\downarrow \frac{S\{t\}}{S[a, \bar{a}]} & \text{aw}\downarrow \frac{S\{f\}}{S\{a\}} & \text{ac}\downarrow \frac{S[a, a]}{S\{a\}} \\
\\
& \text{s} \frac{S([R, T], U)}{S[(R, U), T]} & \text{m} \frac{S[(R, T), (U, V)]}{S([R, U], [T, V])}
\end{array}$$

Fig. 3. System KS

cut. Rules $\text{ai}\downarrow$, $\text{aw}\downarrow$, $\text{ac}\downarrow$ are called *down-rules* and their duals are called *up-rules*. In [2], by a semantic argument, all up-rules were shown to be admissible. By removing them we obtain system KS, shown in Fig. 3, which is complete.

Cut-free sequent systems fulfill the subformula property. Our case is different, because the notions of formula and sequent are merged. System KS does not fulfill a “substructure property” just as sequent systems do not fulfill a “subsequent property”. However, when seen bottom-up, no rule in system KS introduces new atoms. It thus satisfies the main aspect of the subformula property: when given a conclusion of a rule there is only a finite number of premises to choose from. In proof search, for example, the branching of the search tree is finite.

Identity, cut, weakening and contraction are restricted to atoms in system SKS. The general versions of those rules are shown in Fig. 4.

Theorem 5. The rules $i\downarrow$, $w\downarrow$ and $c\downarrow$ are derivable in $\{\text{ai}\downarrow, \text{s}\}$, $\{\text{aw}\downarrow, \text{s}\}$ and $\{\text{ac}\downarrow, \text{m}\}$, respectively. Dually, the rules $i\uparrow$, $w\uparrow$ and $c\uparrow$ are derivable in $\{\text{ai}\uparrow, \text{s}\}$, $\{\text{aw}\uparrow, \text{s}\}$ and $\{\text{ac}\uparrow, \text{m}\}$, respectively.

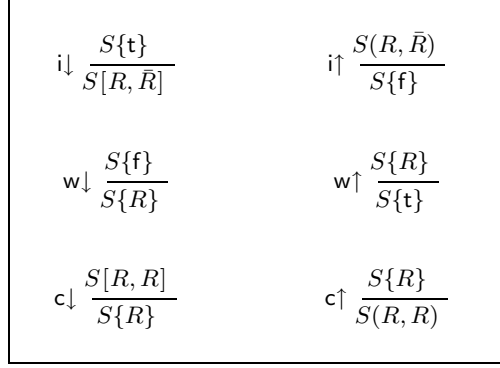


Fig. 4. General identity, weakening, contraction and their duals

Proof. By an easy structural induction on the structure that is cut, weakened or contracted. Details are in [2]. The case for the cut is shown here. A cut introducing the structure (R, T) together with its dual structure $[\bar{R}, \bar{T}]$ is replaced by two cuts on smaller structures:

$$\text{i}\uparrow \frac{S(R, T, [\bar{R}, \bar{T}])}{S\{\mathbf{f}\}} \quad \rightsquigarrow \quad \text{i}\uparrow \frac{S(R, \bar{R})}{S\{\mathbf{f}\}} \quad .$$

$$\begin{array}{c} \frac{S(R, T, [\bar{R}, \bar{T}])}{S\{\mathbf{f}\}} \\ \text{s} \\ \frac{S(R, [\bar{R}, (T, \bar{T})])}{S\{\mathbf{f}\}} \\ \text{s} \\ \frac{S[(R, \bar{R}), (T, \bar{T})]}{S\{\mathbf{f}\}} \\ \text{i}\uparrow \end{array}$$

□

So, while general identity, weakening, contraction and their duals do not belong to **SKS**, they will be freely used in derivations in **SKS** to denote multiple instances of the corresponding rules in **SKS** according to Theorem 5.

Remark 6. Sequent calculus derivations easily correspond to derivations in system **SKS**. For instance, the cut of sequent systems in Gentzen-Schütte form [11]:

$$\text{Cut} \frac{\vdash \Phi, A \quad \vdash \Psi, \bar{A}}{\vdash \Phi, \Psi} \quad \text{corresponds to} \quad \text{i}\uparrow \frac{([\Phi, A], [\Psi, \bar{A}])}{[\Phi, \Psi]} \quad .$$

$$\begin{array}{c} ([\Phi, A], [\Psi, \bar{A}]) \\ \text{s} \\ [\Phi, (A, [\Psi, \bar{A}])] \\ \text{s} \\ [\Phi, \Psi, (A, \bar{A})] \\ \text{i}\uparrow \end{array}$$

4 Cut Elimination

In the calculus of structures, there is more freedom in applying inference rules than in the sequent calculus. While this allows for a richer combinatorial analysis of proofs, it is a significant challenge for cut elimination. During cut elimination,

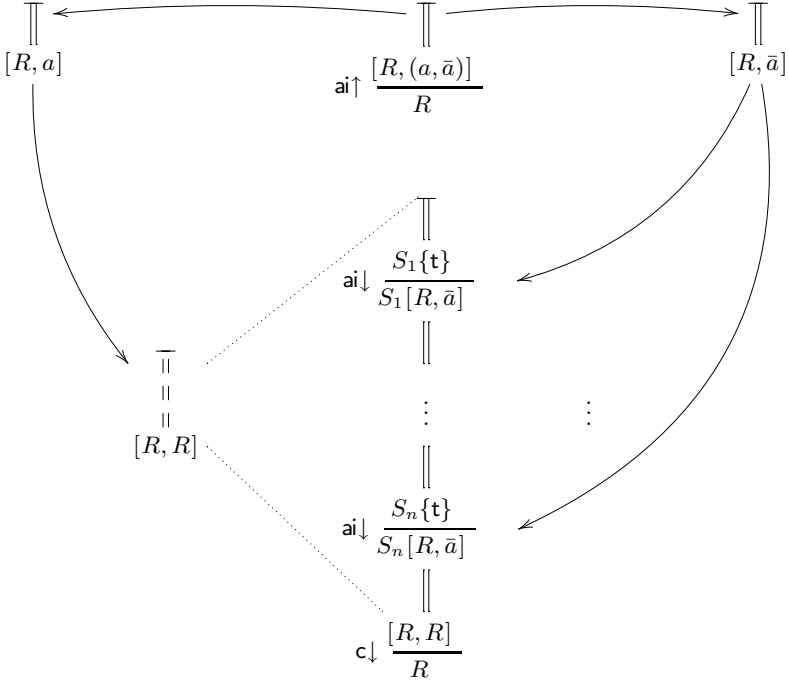


Fig. 5. Elimination of one atomic cut

the sequent calculus allows to get into the crucial situation where on one branch a logical rule applies to the main connective of the eigenformula and on the other branch the corresponding rule applies to the dual connective of the dual eigenformula. In the calculus of structures, rules apply deep inside a context, they are not restricted to main connectives. The methodology of the sequent calculus thus does not apply. For example, one cannot permute the cut over the switch rule. One can generalise the cut in order to permute it over switch, but this requires a case analysis that is far more complicated than in the sequent calculus. Contraction is an even bigger problem. Despite many efforts, no cut elimination procedure along these lines has been found for system SKS.

Two new techniques were developed to eliminate cuts in the calculus of structures. The first is called *decomposition*, and has been used in [6,9] for some systems related to linear logic. Proving termination of decomposition is rather involved [9]. It makes essential use of the exponentials of linear logic which restrict the use of contraction. So far, this technique could not be used for classical logic with its unrestricted contraction. The second technique is called *splitting* [5], and essentially makes available a situation corresponding to the one described above for the sequent calculus. Splitting covers the broadest range of systems in the calculus of structures, it not only applies to the systems mentioned above, but has recently also been applied to system SKS (but the proof is not published

yet). Compared to splitting, the procedure given here is much simpler. In fact, I do not know of any other system with such a simple cut elimination procedure.

In the sequent calculus as well as in sequent-style natural deduction, a derivation is a tree. Seen bottom-up, a cut splits the tree into two branches. To apply a cut, one is forced to split the context among the two branches (in the case of multiplicative context treatment) or to duplicate the context (in the case of additive context treatment). In the calculus of structures, the cut rule does not split the proof.

The crucial idea, illustrated in Fig. 5, is that we can do that during cut elimination. This allows us to plug-in proofs just like in natural deduction: we duplicate the proof above a cut and remove atom a from the copy shown on the left and the atom $\bar{a}$ from the copy shown on the right. We choose one copy, the one on the left in this case, and replace a by R throughout the proof, breaking some instances of identity. They are fixed by substituting the proof on the right. A contraction is applied to obtain a cut-free proof of R .

In contrast to the sequent calculus, the cut is not the only problematic rule in system SKS. The rule $\mathbf{aw}\uparrow$ also induces infinite choice in proof-search. Fortunately, we can not only eliminate the cut rule, but also the other up-rules. Each up-rule individually can be shown to be admissible for system KS. However, since we are going to eliminate the cut anyway, to eliminate rules $\mathbf{aw}\uparrow$ and $\mathbf{ac}\uparrow$ the following lemma is sufficient.

Lemma 7. Each rule in SKS is derivable for identity, cut, switch and its dual rule.

Proof. An instance of $\rho\uparrow \frac{S\{T\}}{S\{R\}}$ can be replaced by $\rho\downarrow \frac{\begin{array}{c} \text{i}\downarrow \frac{S\{T\}}{S(T, [R, \bar{R}])} \\ \text{s} \frac{S(T, [R, \bar{R}])}{S[R, (T, \bar{R})]} \\ \text{p}\downarrow \frac{S[R, (T, \bar{R})]}{S[R, (T, \bar{T})]} \\ \text{i}\uparrow \frac{S[R, (T, \bar{T})]}{S\{R\}} \end{array}}{S\{R\}}$. The

same holds for down-rules. $\square$

When plugging in a derivation in natural deduction, its context has to be strengthened, to fit into the leaf into which it is plugged. Adding to a context in natural deduction is easy, since it is a flat object, a set or a multiset. In the calculus of structures, contexts are more general, nested objects. The following definition is used to strengthen contexts.

Definition 8. Given a derivation Δ , the derivation $S\{\Delta\}$ is obtained as follows:

$$\Delta = \begin{array}{c} \pi' \frac{T}{V} \\ \pi \frac{\quad}{\quad} \\ \vdots \\ \rho' \frac{\quad}{U} \\ \rho \frac{\quad}{R} \end{array} \quad S\{\Delta\} = \begin{array}{c} \pi' \frac{S\{T\}}{S\{V\}} \\ \pi \frac{\quad}{\quad} \\ \vdots \\ \rho' \frac{\quad}{S\{U\}} \\ \rho \frac{\quad}{S\{R\}} \end{array} .$$

Definition 9. An instance of atomic cut is called *shallow* if it is of the following form:

$$\text{ai}\uparrow \frac{[S, (a, \bar{a})]}{S}.$$

Lemma 10. The atomic cut is derivable for shallow atomic cut and switch.

Proof. An easy induction locally replaces an instance of atomic cut by a shallow atomic cut followed by instances of switch. Details are in [1]. $\square$

Lemma 11. Each proof $\prod_{T\{a\}}^{\text{KS}}$ can be transformed into a proof $\prod_{T\{t\}}^{\text{KS}}$.

Proof. Starting with the conclusion, going up in the proof, in each structure we replace the occurrence of a and its copies, that are produced by contractions, by the unit t . Replacements inside the context of any rule instance do not affect the validity of this rule instance. Instances of the rules $\mathbf{m}$ and $\mathbf{s}$ remain valid, also in the case that atom occurrences are replaced inside redex and contractum. Instances of the other rules are replaced by the following derivations:

$$\begin{aligned} \text{ac}\downarrow \frac{S[a, a]}{S\{a\}} &\rightsquigarrow = \frac{S[t, t]}{S\{t\}} \\ &= \frac{S\{f\}}{S([t, t], f)} \\ \text{aw}\downarrow \frac{S\{f\}}{S\{a\}} &\rightsquigarrow = \frac{\text{s} \frac{S\{f\}}{S[t, (t, f)]}}{S\{t\}} \\ \text{ai}\downarrow \frac{S\{t\}}{S[a, \bar{a}]} &\rightsquigarrow = \frac{\text{aw}\downarrow \frac{S\{t\}}{S[t, f]}}{\text{aw}\downarrow \frac{S[t, f]}{S[t, \bar{a}]}}. \end{aligned}$$

$\square$

Properly equipped, we now turn to cut elimination.

Theorem 12. Each proof $\prod_T^{\text{SKS}}$ can be transformed into a proof $\prod_T^{\text{KS}}$.

Proof. By Lemma 7, the only rule left to eliminate is the cut. By Lemma 10, we replace all cuts by shallow cuts. The topmost instance of cut, together with the proof above it, is singled out:

$$\prod_T^{\text{KS} \cup \{\text{ai}\uparrow\}} = \text{ai}\uparrow \frac{\prod_{\Delta \prod_{T\{a\}}^{\text{KS}}}^{\text{KS}}}{R}.$$

Lemma 11 is applied twice on Π to obtain

$$\Pi_1 \left\|_{\text{KS}} \right. \quad \text{and} \quad \Pi_2 \left\|_{\text{KS}} \right. \quad .$$

$$[R, a] \qquad [R, \bar{a}]$$

Starting with the conclusion, going up in proof Π_1 , in each structure we replace the occurrence of a and its copies, that are produced by contractions, by the structure R .

Replacements inside the context of any rule instance do not affect the validity of this rule instance. Instances of the rules $\mathbf{m}$ and $\mathbf{s}$ remain valid, also in the case that atom occurrences are replaced inside redex and contractum. Instances of $\mathbf{ac}\downarrow$ and $\mathbf{aw}\downarrow$ are replaced by their general versions:

$$\mathbf{ac}\downarrow \frac{S[a, a]}{S\{a\}} \quad \rightsquigarrow \quad \mathbf{c}\downarrow \frac{S[R, R]}{S\{R\}}$$

$$\mathbf{aw}\downarrow \frac{S\{f\}}{S\{a\}} \quad \rightsquigarrow \quad \mathbf{w}\downarrow \frac{S\{f\}}{S\{R\}} \quad .$$

Instances of $\mathbf{ai}\downarrow$ are replaced by $S\{\Pi_2\}$:

$$\mathbf{ai}\downarrow \frac{S\{t\}}{S[a, \bar{a}]} \quad \rightsquigarrow \quad \frac{S\{t\}}{S\{\Pi_2\} \left\|_{\text{KS}} \right. S[R, \bar{a}]}$$

The result of this process of substituting Π_2 into Π_1 is a proof Π_3 , from which we build

$$\Pi_3 \left\|_{\text{KS}} \right. \quad \mathbf{c}\downarrow \frac{[R, R]}{R} \quad \Delta \left\|_{\text{KS} \cup \{\mathbf{ai}\downarrow\}} \right. \quad T$$

Proceed inductively downward with the remaining instances of cut. □

5 Conclusion

System SKS seems a good starting point for developing both the proof search as well as the proof normalisation paradigm in one system. Since all up-rules are admissible, it is suitable for proof search as computation. The cut elimination procedure given is simpler than those for sequent calculi. The way in

which proofs are substituted resembles normalisation in natural deduction. This hopefully allows for a computational interpretation in the proof normalisation as computation paradigm.

Of course, a lot of work remains to be done. In the proof search as computation realm, given the admissibility of cut, a suitable notion of *uniform proof* as in [7] should be obtainable. For proof normalisation as computation, natural questions to be considered are strong normalisation and confluence of the cut elimination procedure when imposing as little strategy as possible. Similarly to [8], a term calculus should be developed and its computational meaning be made precise. Intuitionistic logic is a more familiar setting for this, so the possibility of treating intuitionistic logic should be explored.

A natural question is whether this procedure scales to more expressive cases, for example to predicate logic. System SKS_q extends system SKS by first-order quantifiers [1]. There, cut elimination is proved via a translation to the sequent calculus. The procedure presented here does not appear to easily scale to system SKS_q . The problem, which does not occur in shallow inference systems like sequent calculus or natural deduction, are existential quantifiers in the context of a cut which bind variables both in a and $\bar{a}$. The procedure easily extends to *closed* atomic cuts, that is, cuts where the eigenformula is an atom prefixed by quantifiers that bind all its variables. The question then is how to reduce general cuts to closed atomic cuts. If this problem were solved, then the procedure would scale to predicate logic. Hopefully this will lead to a cut elimination procedure for predicate logic, which is simpler than other cut elimination procedures, as happened for propositional logic.

Acknowledgements

This work has been supported by the DFG Graduiertenkolleg 334. I would like to thank Alessio Guglielmi, for inspiration, and Alwen Fernanto Tiu and Michel Parigot, for comments which helped simplifying an earlier version of the cut elimination procedure. Alessio Guglielmi, Charles Stewart and Lutz Straburger carefully read previous drafts of this work and prompted improvements.

Web Site

Information about the calculus of structures is available from:

<http://www.wv.inf.tu-dresden.de/~guglielm/Research/>.

References

1. Kai Brännler. Locality for classical logic. Technical Report WV-02-15, Dresden University of Technology, 2002.
Available at <http://www.wv.inf.tu-dresden.de/~kai/LocalityClassical.pdf>.
2. Kai Brännler and Alwen Fernanto Tiu. A local system for classical logic. In R. Nieuwenhuis and A. Voronkov, editors, *LPAR 2001*, volume 2250 of *Lecture Notes in Artificial Intelligence*, pages 347–361. Springer-Verlag, 2001.

3. Paola Bruscoli. A purely logical account of sequentiality in proof search. In Peter J. Stuckey, editor, *Logic Programming, 18th International Conference*, volume 2401 of *Lecture Notes in Artificial Intelligence*, pages 302–316. Springer-Verlag, 2002.
4. Jean Gallier. Constructive logics. Part I: A tutorial on proof systems and typed λ -calculi. *Theoretical Computer Science*, 110:249–339, 1993.
5. Alessio Guglielmi. A system of interaction and structure. Technical Report WV-02-10, 2002. Available at <http://www.wv.inf.tu-dresden.de/~guglielm/Research/Gug/Gug.pdf>.
6. Alessio Guglielmi and Lutz Straßburger. Non-commutativity and MELL in the calculus of structures. In L. Fribourg, editor, *CSL 2001*, volume 2142 of *Lecture Notes in Computer Science*, pages 54–68. Springer-Verlag, 2001.
7. Dale Miller, Gopalan Nadathur, Frank Pfenning, and Andre Scedrov. Uniform proofs as a foundation for logic programming. *Annals of Pure and Applied Logic*, 51:125–157, 1991.
8. M. Parigot. $\lambda\mu$ -calculus: an algorithmic interpretation of classical natural deduction. In *LPAR 1992*, volume 624 of *Lecture Notes in Computer Science*, pages 190–201. Springer-Verlag, 1992.
9. Lutz Straßburger. MELL in the calculus of structures. Technical Report WV-2001-03, Dresden University of Technology, 2001. Accepted by TCS. Available at <http://www.ki.inf.tu-dresden.de/~lutz/els.pdf>.
10. Lutz Straßburger. A local system for linear logic. In Matthias Baaz and Andrei Voronkov, editors, *Logic for Programming, Artificial Intelligence, and Reasoning, LPAR 2002*, volume 2514 of *LNAI*, pages 388–402. Springer-Verlag, 2002.
11. Anne Sjerp Troelstra and Helmut Schwichtenberg. *Basic Proof Theory*. Cambridge University Press, 1996.

A Non-commutative Extension of MELL

Alessio Guglielmi and Lutz Straßburger

Technische Universität Dresden

Fakultät Informatik - 01062 Dresden - Germany

{Alessio.Guglielmi,Lutz.Strassburger}@Inf.TU-Dresden.DE

Abstract. *We extend multiplicative exponential linear logic (MELL) by a non-commutative, self-dual logical operator. The extended system, called NEL, is defined in the formalism of the calculus of structures, which is a generalisation of the sequent calculus and provides a more refined analysis of proofs. We should then be able to extend the range of applications of MELL by modelling a broad notion of sequentiality and providing new properties of proofs. We show some proof theoretical results: decomposition and cut elimination. The new operator represents a significant challenge: to get our results we use here for the first time some novel techniques, which constitute a uniform and modular approach to cut elimination, contrary to what is possible in the sequent calculus.*

1 Introduction

Non-commutative logical operators have a long tradition [12, 22, 2, 13, 16, 3], and their proof theoretical properties have been studied in the sequent calculus [7] and in proof nets [8]. Recent research has shown that the sequent calculus is not adequate to deal with very simple forms of non-commutativity [9, 10, 21]. On the other hand, proof nets are not ideal for dealing with exponentials and additives, which are desirable for getting good computational power.

In this paper we show a logical system that joins a simple form of non-commutativity with commutative multiplicatives and exponentials. This is done in the formalism of the calculus of structures [9, 10], which overcomes the difficulties encountered in the sequent calculus and in proof nets. Structures are expressions intermediate between formulae and sequents, and in fact they unify those two latter entities into a single one, thereby allowing more control over mutual dependencies of logical relations.

We perform a proof theoretical analysis for cut elimination, with new tools, and we explore some further important properties, which are not available in more traditional settings and which we can collectively regard as ‘modularity’. Despite the complexities of the proof theoretical investigation, the system obtained is very simple. This paper contributes the following new results:

- 1 We define a propositional logical system, called NEL (Non-commutative exponential linear logic), which extends MELL (multiplicative exponential linear logic [8]) by a non-commutative, self-dual logical operator called *seq*. This system, which was first imagined in [10], is conservative over MELL augmented by the mix and nullary mix rules [1, 6]. System NEL can be immediately understood by anybody acquainted with the sequent calculus, and is aimed at the same range of applications as MELL. In nearly all computer science languages, sequential composition plays a fundamental role,

and it is therefore important to address it in a direct way, in logical representations of those languages. Perhaps surprisingly, parallel composition has been much easier to deal with, due to its commutative nature, which is more similar to the typical nature of traditional logics. The addition of seq opens new syntactic possibilities, for example in dealing with process algebras. It has been used already, in a purely multiplicative setting, to model CCS's prefixing [5]. Furthermore, we show a class of equivalent extensions of NEL, which all enjoy the subformula property. This, together with the finer detail in derivations achieved by the calculus of structures, provides much greater flexibility, as witnessed by the proof theoretical properties mentioned below.

- 2 We prove for NEL a property called *decomposition* (first pioneered in [10, 19]): we can transform every derivation into an equivalent one, composed of seven derivations carried into seven *disjoint* subsystems of NEL. We can study small subsystems of NEL in isolation and then compose them together with considerable more freedom than in the sequent calculus, where, for example, contraction can not be isolated in a derivation. Decomposition is made available in the calculus of structures by exploiting a new top-down symmetry of derivations. Since it is a basic compositional result, we expect applications to be very broad in range; we are especially excited about the possibilities in the semantics of derivations.
- 3 We prove cut elimination for NEL by use of decomposition and a new technique that we call *splitting*. In the calculus of structures the traditional methods for proving cut elimination fail, due to the more general applicability of inference rules. The deep reason for this is in how the calculus deals with associativity. Splitting theorems are a uniform means of recovering control over the way logical operators associate; they allow us to manage the complex inductions required. The cut elimination argument becomes modular, because we can reduce the cut rule to several more primitive inference rules, each of which is separately shown admissible by way of splitting. Only one of these rules (an atomic form of cut) is infinitary, all the others enjoy the subformula property and can be used to extend the system without affecting provability. This result should be handy for software analysis and verification.

The points above correspond, respectively, to Sections 2, 3 and 4. Readers who are not interested in the proof theory of system NEL can just read Section 2.

Other systems extending linear logic with non-commutative operators are studied in [3, 18]. These are more traditional systems in the sequent calculus, for which a more limited proof theory can be developed. The calculus of structures allows us to design a much simpler logic, as witnessed by the fact that we have just one self-dual non-commutative operator instead of two dual ones.

It is worth noting that every system that can be expressed in the one-sided sequent calculus can be trivially expressed in the calculus of structures, but the vice versa is not true. The results in this paper help us to establish the calculus of structures as a natural choice for logical systems aimed at computer science. We showed in [10] that the sequent calculus suffers from excessive restrictions, which are not apparent in the traditional systems of classical and intuitionistic

logics, but which start to appear in linear logic and are more and more evident when issues such as non-commutativity, locality of inference rules, and various forms of modularity are taken into account. The calculus of structures was in fact conceived, in [9], as a way to overcome the limitations of the sequent calculus in dealing with non-commutativity. Our calculus has later been used successfully in [19] for defining pure MELL and showing decomposition and cut elimination for it. In [4] a completely local definition of classical logic is shown: in that system, not only the cut rule, but also contraction is atomic.

The calculus of structures essentially introduces two new ideas: 1) it makes derivations top-down symmetric and 2) it allows inference rules to be applied anywhere deep inside structures. We are showing, in this and other papers, that it is possible to produce a rich proof theory in our calculus. This formalism is less dependent than the sequent calculus or natural deduction on the original idiosyncrasies of classical (and intuitionistic) logic, and it is actually designed with notions of locality, atomicity and modularity in mind. For these reasons we promote the calculus of structures as a worthy tool for syntactic investigations related to computer science languages.

In the following sections some proof theory is developed for system NEL. We stress the fact that the methods used are general. As stated above, many techniques in this paper are new, but we tested them privately on the systems that have already been studied, namely BV, MELL and classical logic, and in some systems that we are currently investigating, like full linear logic, also in its entirely atomic presentation [20].

The results in this paper are shown in detail in [11].

2 The System

We call *calculus* a formalism, like natural deduction or the sequent calculus, for specifying logical systems. We say (*formal*) *system* to indicate a collection of inference rules in a given calculus.

A system in our calculus requires a language of *structures*, which are intermediate expressions between formulae and sequents. We now define the language for system NEL and its variants. Intuitively, $[S_1, \dots, S_h]$ corresponds to a sequent $\vdash S_1, \dots, S_h$ in linear logic, whose formulae are essentially connected by pars, subject to commutativity (and associativity). The structure $(S_1, \dots, S_h)$ corresponds to the associative and commutative times connection of $S_1, \dots, S_h$. The structure $\langle S_1; \dots; S_h \rangle$ is associative and *non-commutative*: this corresponds to the new logical operator, called *seq*, that we add to those of MELL.

For reasons explained in [9, 10], dealing with *seq* involves adding the rules *mix* and its nullary version *mix0* (see [1, 6, 14]):

$$\text{mix} \frac{\vdash \Phi \quad \vdash \Psi}{\vdash \Phi, \Psi} \quad \text{and} \quad \text{mix0} \frac{}{\vdash} .$$

This has the effect of collapsing the multiplicative units 1 and $\perp$: we will only have one unit $\circ$ common to par, times and *seq*. Please notice that *mix* and *mix0*

are not an artefact of the calculus of structures. But, as shown by Retoré in [14], they are required when using a self-dual non-commutative connective.

2.1 Definition There are countably many *positive* and *negative atoms*. They, positive or negative, are denoted by $a, b, \dots$. *Structures* are denoted by S, P, Q, R, T, U, V and X . The structures of the *language* NEL are generated by

$$S ::= a \mid \circ \mid \underbrace{[S, \dots, S]}_{>0} \mid \underbrace{(S, \dots, S)}_{>0} \mid \underbrace{\langle S; \dots; S \rangle}_{>0} \mid ?S \mid !S \mid \bar{S} \quad ,$$

where $\circ$, the *unit*, is not an atom; $[S_1, \dots, S_h]$ is a *par structure*, $(S_1, \dots, S_h)$ is a *times structure*, $\langle S_1; \dots; S_h \rangle$ is a *seq structure*, $?S$ is a *why-not structure* and $!S$ is an *of-course structure*; $\bar{S}$ is the *negation* of the structure S . Structures with a hole that does not appear in the scope of a negation are denoted by $S\{ \}$. The structure R is a *substructure* of $S\{R\}$, and $S\{ \}$ is its *context*. We simplify the indication of context in cases where structural parentheses fill the hole exactly: for example, $S[R, T]$ stands for $S\{[R, T]\}$.

Structures come with equational theories establishing some basic, decidable algebraic laws by which structures are indistinguishable. These are analogous to the laws of associativity, commutativity, idempotency, and so on, usually imposed on sequents. The difference is that we merge the notions of formula and sequent, and we extend the equations to formulae. We will show these equational laws together with the inference rules.

2.2 Definition An (*inference*) *rule* is any scheme $\rho \frac{T}{R}$, where ρ is the *name* of the rule, T is its *premise* and R is its *conclusion*; R or T , but not both, may be missing. Rule names are denoted by ρ . A (*formal*) *system*, denoted by $\mathcal{S}$, is a set of rules. A *derivation* in a system $\mathcal{S}$ is a finite chain of instances of rules of $\mathcal{S}$, and is denoted by Δ ; a derivation can consist of just one structure. The topmost structure in a derivation is called its *premise*; the bottommost structure is called *conclusion*. A derivation Δ whose premise is T , conclusion is R , and whose rules are in $\mathcal{S}$ is denoted by $\Delta \parallel_{\mathcal{S}}^{\frac{T}{R}}$.

The typical inference rules are of the kind $\rho \frac{S\{T\}}{S\{R\}}$. This rule scheme ρ specifies that if a structure matches R , in a context $S\{ \}$, it can be rewritten as specified by T , in the same context $S\{ \}$ (or vice versa if one reasons top-down). A rule corresponds to implementing in the formal system *any axiom* $T \Rightarrow R$, where $\Rightarrow$ stands for the implication we model in the system, in our case linear implication. The case where the context is empty corresponds to the sequent calculus. For example, the linear logic sequent calculus rule $\otimes \frac{\vdash A, \Phi \quad \vdash B, \Psi}{\vdash A \otimes B, \Phi, \Psi}$ could be simulated easily in the calculus of structures by the rule $\otimes' \frac{(I, [A, \Phi], [B, \Psi])}{(I, [(A, B), \Phi, \Psi])}$, where Φ and Ψ stand for multisets of formulae or their corresponding par structures. The structure I stands for the times structure of the other hypotheses in the derivation tree. More precisely, any sequent calculus derivation

$$\begin{array}{c}
 \vdash \Gamma_1 \quad \dots \quad \vdash \Gamma_{i-1} \quad \otimes \quad \frac{\vdash A, \Phi \quad \vdash B, \Psi}{\vdash A \otimes B, \Phi, \Psi} \quad \vdash \Gamma_{i+1} \quad \dots \quad \vdash \Gamma_h \\
 \hline
 \Delta \\
 \hline
 \vdash \Sigma
 \end{array}$$

containing the $\otimes$ rule can be simulated by

$$\begin{array}{c}
 \otimes' \quad \frac{(\Gamma_1, \dots, \Gamma_{i-1}, [A, \Phi], [B, \Psi], \Gamma_{i+1}, \dots, \Gamma_h)}{(\Gamma_1, \dots, \Gamma_{i-1}, [(A, B), \Phi, \Psi], \Gamma_{i+1}, \dots, \Gamma_h)} \\
 \Delta \parallel \\
 \Sigma
 \end{array}$$

in the calculus of structures, where Γ_j , A , B , Φ , Ψ , Δ and Σ are obtained from their counterparts in the sequent calculus by the obvious translation. This means that by this method every system in the one-sided sequent calculus can be ported trivially to the calculus of structures.

Of course, in the calculus of structures, rules could be used as axioms of a generic Hilbert system, where there is no special, structural relation between T and R : then all the good proof theoretical properties of sequent systems would be lost. We will be careful to design rules in a way that is conservative enough to allow us to prove cut elimination, and such that they possess the subformula property.

In our systems, rules come in pairs, $\rho \downarrow \frac{S\{T\}}{S\{R\}}$ (down version) and $\rho \uparrow \frac{S\{\bar{R}\}}{S\{\bar{T}\}}$ (up version). Sometimes rules are self-dual, i.e., the up and down version are identical, in which case we omit the arrows. This duality derives from the duality between $T \Rightarrow R$ and $\bar{R} \Rightarrow \bar{T}$. We will be able to get rid of the up rules without affecting provability—after all, $T \Rightarrow R$ and $\bar{R} \Rightarrow \bar{T}$ are equivalent statements in many logics. Remarkably, the cut rule reduces into several up rules, and this makes for a modular decomposition of the cut elimination argument because we can eliminate up rules one independently from the other.

Let us now define system NEL by starting from a top-down symmetric variation, that we call SNEL. It is made by two sub-systems that we will call conventionally *interaction* and *structure*. The interaction fragment deals with negation, i.e., duality. It corresponds to identity and cut in the sequent calculus. In our calculus these rules become mutually top-down symmetric and both can be reduced to their atomic counterparts.

The structure fragment corresponds to logical and structural rules in the sequent calculus; it defines the logical operators. Differently from the sequent calculus, the operators need not be defined in isolation, rather complex contexts can be taken into consideration. In the following system we consider *pairs* of logical relations, one inside the other.

Associativity	Commutativity
$[\vec{R}, [\vec{T}]] = [\vec{R}, \vec{T}]$	$[\vec{R}, \vec{T}] = [\vec{T}, \vec{R}]$
$(\vec{R}, (\vec{T})) = (\vec{R}, \vec{T})$	$(\vec{R}, \vec{T}) = (\vec{T}, \vec{R})$
$\langle \vec{R}; \langle \vec{T} \rangle; \vec{U} \rangle = \langle \vec{R}; \vec{T}; \vec{U} \rangle$	
	Negation
Singleton	$\bar{\circ} = \circ$
$[R] = (R) = \langle R \rangle = R$	$\overline{[R, T]} = (\bar{R}, \bar{T})$
	$\overline{(R, T)} = [\bar{R}, \bar{T}]$
Units	$\overline{\langle R, T \rangle} = \langle \bar{R}; \bar{T} \rangle$
$[\circ, \vec{R}] = [\vec{R}]$	$\overline{?R} = !\bar{R}$
$(\circ, \vec{R}) = (\vec{R})$	$\overline{!R} = ?\bar{R}$
$\langle \circ; \vec{R} \rangle = \langle \vec{R} \rangle$	$\bar{\bar{R}} = R$
$\langle \vec{R}; \circ \rangle = \langle \vec{R} \rangle$	
Exponentials	Contextual Closure
$? \circ = ! \circ = \circ$	if $R = T$
$??R = ?R$	then $S\{R\} = S\{T\}$
$!!R = !R$	

$\text{ai} \downarrow \frac{S\{\circ\}}{S[a, \bar{a}]}$	$\text{ai} \uparrow \frac{S(a, \bar{a})}{S\{\circ\}}$
Interaction	
Structure	
$\text{s} \frac{S([R, U], T)}{S[(R, T), U]}$	
$\text{q} \downarrow \frac{S(\langle [R, U]; [T, V] \rangle)}{S[\langle R; T \rangle, \langle U; V \rangle]}$	$\text{q} \uparrow \frac{S(\langle R; U \rangle, \langle T; V \rangle)}{S(\langle R, T \rangle; \langle U, V \rangle)}$
$\text{p} \downarrow \frac{S\{! [R, T]\}}{S[! R, ?T]}$	$\text{p} \uparrow \frac{S(? R, !T)}{S\{?(R, T)\}}$
core	
non-core	
$\text{w} \downarrow \frac{S\{\circ\}}{S\{?R\}}$	$\text{w} \uparrow \frac{S\{!R\}}{S\{\circ\}}$
$\text{b} \downarrow \frac{S\{?R, R\}}{S\{?R\}}$	$\text{b} \uparrow \frac{S\{!R\}}{S\{!(R, R)\}}$

Fig. 1 Left: *Syntactic equivalence* = Right: *System SNEL*

2.3 Definition The structures of the language NEL are equivalent modulo the relation $=$, defined at the left of Fig. 1. There, $\vec{R}$, $\vec{T}$ and $\vec{U}$ stand for finite, non-empty sequences of structures (sequences may contain ‘,’ or ‘;’ separators as appropriate in the context). At the right of the figure, *system SNEL* is shown (*symmetric non-commutative exponential linear logic*). The rules $\text{ai} \downarrow$, $\text{ai} \uparrow$, s , $\text{q} \downarrow$, $\text{q} \uparrow$, $\text{p} \downarrow$, $\text{p} \uparrow$, $\text{w} \downarrow$, $\text{w} \uparrow$, $\text{b} \downarrow$ and $\text{b} \uparrow$, are called respectively *atomic interaction*, *atomic cut*, *switch*, *seq*, *coseq*, *promotion*, *copromotion*, *weakening*, *coweakening*, *absorption* and *coabsorption*. The *down fragment* of SNEL is $\{\text{ai} \downarrow, \text{s}, \text{q} \downarrow, \text{p} \downarrow, \text{w} \downarrow, \text{b} \downarrow\}$, the *up fragment* is $\{\text{ai} \uparrow, \text{s}, \text{q} \uparrow, \text{p} \uparrow, \text{w} \uparrow, \text{b} \uparrow\}$.

There is a straightforward two-way correspondence between structures not involving *seq* and formulae of MELL: for example $![(?a, b), \bar{c}, !\bar{d}]$ corresponds to $!((?a \otimes b) \wp c^\perp \wp !d^\perp)$, and vice versa. Units are mapped into $\circ$, since $1 \equiv \perp$, when *mix* and *mix0* are added to MELL. System SNEL is just the merging of systems SBV and SELS shown in [9, 10, 19]; there one can find details on the correspondence between our systems and linear logic. The reader can check that the equations in Fig. 1 correspond to logical equivalences in MELL, disregarding *seq*. In particular, $!A \multimap !!A$ and $!!A \multimap !A$ for every MELL formula A , and dually for $?$. The rules s , $\text{q} \downarrow$ and $\text{q} \uparrow$ are the same as in pomset logic viewed as a calculus of cographs [17].

All equations are typical of a sequent calculus presentation, save those for units, exponentials and contextual closure. Contextual closure just corresponds to equivalence being a congruence, it is a necessary ingredient of the calculus of

structures. All other equations can be removed and replaced by rules, as in the sequent calculus. This might prove necessary for certain applications. For our purposes, this setting makes for a much more compact presentation, at a more effective abstraction level.

Negation is involutive and can be pushed to atoms; it is convenient always to imagine it directly over atoms. Please note that negation does not swap arguments of seq, as happens in the systems of Lambek and Abrusci-Ruet. The unit $\circ$ is self-dual and common to par, times and seq. One may think of it as a convenient way of expressing the empty sequence. Rules become very flexible in the presence of the unit. For example, the following notable derivation is valid:

$$\begin{array}{c} \text{q}\uparrow \frac{(a, b)}{\langle a; \circ \rangle} \\ \text{q}\downarrow \frac{\langle a; \circ \rangle}{[a, b]} \end{array} = \begin{array}{c} \text{q}\uparrow \frac{(\langle a; \circ \rangle, \langle \circ; b \rangle)}{\langle [a, \circ]; [\circ, b] \rangle = \langle \langle a, \circ \rangle; (\circ, b) \rangle} \\ \text{q}\downarrow \frac{\langle [a, \circ]; [\circ, b] \rangle}{[\langle a; \circ \rangle, \langle \circ; b \rangle]} \end{array} .$$

Each inference rule in Fig. 1 corresponds to a linear implication that is sound in MELL plus mix and mix0. For example, promotion corresponds to the implication $!(R \wp T) \multimap (!R \wp ?T)$. Notice that interaction and cut are atomic in SNEL; we can define their general versions as follows.

2.4 Definition The following rules are called *interaction* and *cut*:

$$\text{i}\downarrow \frac{S\{\circ\}}{S[R, \bar{R}]} \quad \text{and} \quad \text{i}\uparrow \frac{S(R, \bar{R})}{S\{\circ\}} ,$$

where R and $\bar{R}$ are called *principal structures*.

The sequent calculus rule cut $\frac{\vdash A, \Phi \quad \vdash A^\perp, \Psi}{\vdash \Phi, \Psi}$ is realised as

$$\begin{array}{c} \text{s} \frac{([A, \Phi], [\bar{A}, \Psi])}{[(A, \Phi), (\bar{A}, \Psi)]} \\ \text{s} \frac{[(A, \Phi), (\bar{A}, \Psi)]}{[(A, \bar{A}), \Phi, \Psi]} \\ \text{i}\uparrow \frac{[(A, \bar{A}), \Phi, \Psi]}{[\Phi, \Psi]} , \end{array}$$

where Φ and Ψ stand for multisets of formulae or their corresponding par structures. Notice how the tree shape of derivations in the sequent calculus is realised by making use of times structures: in the derivation above, the premise corresponds to the two branches of the cut rule. For this reason, in the calculus of structures rules are allowed to access structures deeply nested into contexts.

The cut rule in the calculus of structures can mimic the classical cut rule in the sequent calculus in its realisation of transitivity, but it is much more general. We believe a good way of understanding it is in thinking of the rule as being about lemmas *in context*. The sequent calculus cut rule generates a lemma valid in the most general context; the new cut rule does the same, but the lemma only affects the limited portion of structure that can interact with it.

We easily get the next two propositions, which say: 1) The interaction and cut rules can be reduced into their atomic forms—note that in the sequent calculus it is possible to reduce interaction to atomic form, but not cut. 2) The cut rule is as powerful as the whole up fragment of the system, and vice versa.

$\circ\downarrow \frac{}{\circ}$	$\text{ai}\downarrow \frac{S\{\circ\}}{S[a, \bar{a}]}$	$\text{s} \frac{S([R, U], T)}{S[(R, T), U]}$	$\text{q}\downarrow \frac{S(\langle [R, U]; [T, V] \rangle)}{S[\langle R; T \rangle, \langle U; V \rangle]}$
	$\text{p}\downarrow \frac{S\{! [R, T]\}}{S[! R, ? T]}$	$\text{w}\downarrow \frac{S\{\circ\}}{S\{? R\}}$	$\text{b}\downarrow \frac{S[? R, R]}{S\{? R\}}$

Fig. 2 System NEL

2.5 Definition A rule ρ is *derivable* in the system $\mathcal{S}$ if $\rho \notin \mathcal{S}$ and for every instance $\rho \frac{T}{R}$ there exists a derivation $\frac{T}{R} \parallel \mathcal{S}$. The systems $\mathcal{S}$ and $\mathcal{S}'$ are *strongly equivalent* if for every derivation $\frac{T}{R} \parallel \mathcal{S}$ there exists a derivation $\frac{T}{R} \parallel \mathcal{S}'$, and vice versa.

2.6 Proposition The rule $\text{i}\downarrow$ is derivable in $\{\text{ai}\downarrow, \text{s}, \text{q}\downarrow, \text{p}\downarrow\}$, and, dually, the rule $\text{i}\uparrow$ is derivable in the system $\{\text{ai}\uparrow, \text{s}, \text{q}\uparrow, \text{p}\uparrow\}$.

Proof Induction on principal structures. We show the inductive cases for $\text{i}\uparrow$:

$$\begin{array}{c} \frac{S(P, Q, [\bar{P}, \bar{Q}])}{\text{s} \frac{S(Q, [(P, \bar{P}), \bar{Q}])}{\text{s} \frac{S[(P, \bar{P}), (Q, Q)]}{\text{i}\uparrow \frac{S(Q, \bar{Q})}{S\{\circ\}}}}} \\ \text{i}\uparrow \frac{S(Q, \bar{Q})}{S\{\circ\}} \end{array}, \quad \begin{array}{c} \frac{S(\langle (P; Q), \langle \bar{P}; \bar{Q} \rangle)}{\text{q}\uparrow \frac{S(\langle (P, \bar{P}); (Q, \bar{Q}) \rangle)}{\text{i}\uparrow \frac{S(Q, \bar{Q})}{S\{\circ\}}}}} \\ \text{i}\uparrow \frac{S(Q, \bar{Q})}{S\{\circ\}} \end{array} \quad \text{and} \quad \begin{array}{c} \frac{S(? P, ! \bar{P})}{\text{p}\uparrow \frac{S\{?(P, \bar{P})\}}{\text{i}\uparrow \frac{S\{\circ\}}{S\{? R\}}}} \\ \text{i}\uparrow \frac{S\{\circ\}}{S\{? R\}} \end{array}.$$

The cases for $\text{i}\downarrow$ are dual. $\square$

2.7 Proposition Each rule $\rho\uparrow$ in SNEl is derivable in $\{\text{i}\downarrow, \text{i}\uparrow, \text{s}, \rho\downarrow\}$, and, dually, each rule $\rho\downarrow$ in SNEl is derivable in the system $\{\text{i}\downarrow, \text{i}\uparrow, \text{s}, \rho\uparrow\}$.

Proof Each instance $\rho\uparrow \frac{S\{T\}}{S\{R\}}$ can be replaced by $\text{i}\uparrow \frac{S\{T\}}{S\{R\}}$. $\square$

In the calculus of structures, we call *core* the set of rules, other than atomic interaction and cut, used to reduce interaction and cut to atomic form. Rules, other than interaction and cut, that are not in the core are called *non-core*.

2.8 Definition The *core* of SNEl is $\{\text{s}, \text{q}\downarrow, \text{q}\uparrow, \text{p}\downarrow, \text{p}\uparrow\}$, denoted by SNEl_c.

System SNEl is top-down symmetric, and the properties we saw are also symmetric. Provability is an asymmetric notion: we want to observe the possible conclusions that we can obtain from a unit premise. We now break the top-down symmetry by adding an inference rule with no premise, and we join this logical axiom to the down fragment of SNEl.

2.9 Definition The following rule is called *unit*: $\circ\downarrow \frac{}{\circ}$. System NEL is shown in Fig. 2.

As an immediate consequence of Propositions 2.6 and 2.7 we get:

2.10 Theorem The systems $\text{NEL} \cup \{\text{i}\uparrow\}$ and $\text{SNEl} \cup \{\circ\downarrow\}$ are strongly equivalent.

2.11 Definition A derivation with no premise is called a *proof*, denoted by Π . A system $\mathcal{S}$ *proves* R if there is in the system $\mathcal{S}$ a proof Π whose conclusion is R , written $\Pi \Vdash_{\mathcal{S}} R$. We say that a rule ρ is *admissible* for the system $\mathcal{S}$ if $\rho \notin \mathcal{S}$ and for every proof $\frac{R}{\vdash_{\mathcal{S} \cup \{\rho\}}}$ there exists a proof $\frac{R}{\vdash_{\mathcal{S}}}$. Two systems are *equivalent* if they prove the same structures.

Except for cut and coweakening, systems SNEL and NEL enjoy a subformula property (which we treat as an asymmetric property, by going from conclusion to premise): premises are made of substructures of the conclusions.

To get cut elimination, so as to have a system whose rules all enjoy the subformula property, we could just get rid of $\text{ai}\uparrow$ and $\text{w}\uparrow$, by proving their admissibility for the other rules. But we can do more than that: the whole up fragment of SNEL, except for s (which also belongs to the down fragment), is admissible. This entails a *modular* scheme for proving cut elimination. In Sections 3 and 4 we will sketch a proof of the cut elimination theorem:

2.12 Theorem *System NEL is equivalent to every subsystem of $\text{SNEL} \cup \{\circ\downarrow\}$ which contains NEL.*

2.13 Corollary *The rule $\text{i}\uparrow$ is admissible for system NEL.*

Proof Immediate from Theorems 2.10 and 2.12. $\square$

Any implication $T \multimap R$, i.e. $[\bar{T}, R]$, is connected to derivability by:

2.14 Corollary *For any two structures T and R , there is a proof $\frac{T}{\vdash_{\text{NEL}} [\bar{T}, R]}$ iff there is a derivation $\frac{R}{\vdash_{\text{SNEL}}}$.*

Consistency follows as usual and can be proved by way of the same technique used in [10]. It is also easy to prove that system NEL is a conservative extension of MELL plus mix and $\text{mix}0$ (see [9]). The locality properties shown in [10, 19] still hold in this system, of course. In particular, the promotion rule is local, as opposed to the same rule in the sequent calculus.

3 Decomposition

The new top-down symmetry of derivations in the calculus of structures allows to study properties that are not observable in the sequent calculus. The most remarkable results so far are decomposition theorems. In general, a decomposition theorem says that a given system $\mathcal{S}$ can be divided into n pairwise disjoint subsystems $\mathcal{S}_1, \dots, \mathcal{S}_n$ such that every derivation Δ in system $\mathcal{S}$ can be rearranged as composition of n derivations $\Delta_1, \dots, \Delta_n$, where Δ_i uses only rules of $\mathcal{S}_i$, for every $1 \leq i \leq n$.

For system SNEL, we have two such results, which both state a decomposition of every derivation into seven subsystems. They can be stated together as follows:

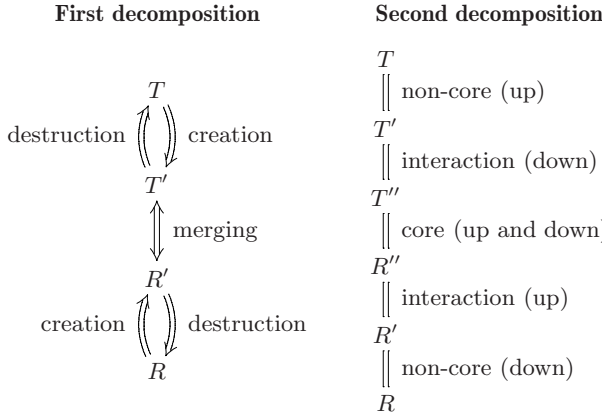


Fig. 3 Readings of the decomposition theorem

3.1 Theorem For every derivation $\frac{T}{R} \parallel_{\text{SNEL}}$ there exist derivations

$$\begin{array}{ccc}
 \frac{T}{R} & & \frac{T}{R} \\
 \parallel \{b\uparrow\} & & \parallel \{b\uparrow\} \\
 T_1 & & T'_1 \\
 \parallel \{w\downarrow\} & & \parallel \{w\downarrow\} \\
 T_2 & & T'_2 \\
 \parallel \{a\downarrow\} & & \parallel \{a\downarrow\} \\
 T_3 & & T'_3 \\
 \parallel_{\text{SNELc}} & \text{and} & \parallel_{\text{SNELc}} \\
 R_3 & & R'_3 \\
 \parallel \{a\uparrow\} & & \parallel \{a\uparrow\} \\
 R_2 & & R'_2 \\
 \parallel \{w\uparrow\} & & \parallel \{w\downarrow\} \\
 R_1 & & R'_1 \\
 \parallel \{b\downarrow\} & & \parallel \{b\downarrow\} \\
 R & & R
 \end{array}$$

for some structures $T_1, T_2, T_3, R_1, R_2, R_3$ and $T'_1, T'_2, T'_3, R'_1, R'_2, R'_3$.

The first decomposition can also be read as a decomposition of a derivation into three parts, which can be called *creation*, where the size of the structure is increased, *merging*, where the size of the structure does not change, and *destruction*, where the size of the structure is decreased. The merging part is in the middle of the derivation and (depending on your preferred reading of a derivation) creation and destruction are at the top and at the bottom, as depicted at the left in Fig. 3. In system SNEL the merging part contains the rules $s, q\downarrow, q\uparrow, p\downarrow$ and $p\uparrow$, which coincides with the core. In the top-down reading of a derivation, the creation part contains the rules $b\uparrow, w\downarrow$ and $a\downarrow$, and the destruction part consists of $b\downarrow, w\uparrow$ and $a\uparrow$. In the bottom-up reading, creation and destruction are exchanged.

Such a decomposition is not restricted to system SNEL. It also holds for other systems in the calculus of structures, including systems SBV and SELS [10],

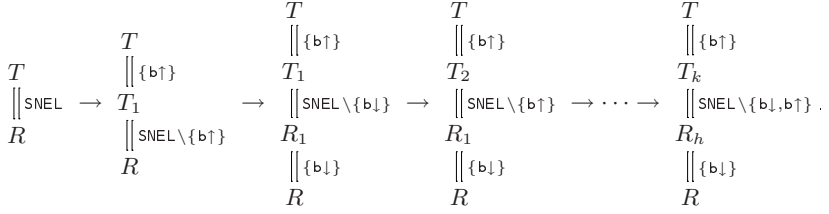


Fig. 4 *Permuting $b\uparrow$ up and $b\downarrow$ down*

classical logic [4] and full propositional linear logic.

The second decomposition in Theorem 3.1 states that in any derivation we can separate five homogeneous subsystems, as shown at the right of Fig. 3. In particular, we can separate the non-core part of the system from the core.

We prove the two decomposition statements similarly. The first step is the separation of absorption. For this, the instances of $b\uparrow$ are permuted over all the other rules. The only problematic case is when

$$\begin{array}{c} \text{p}\downarrow \\ \text{b}\uparrow \end{array} \frac{S\{! [R, T]\}}{S[! R, ?T]} \quad \text{is replaced by} \quad \begin{array}{c} \text{b}\uparrow \\ \text{p}\downarrow \end{array} \frac{S\{! [R, T]\}}{S(! [R, T], [R, T])} \quad \begin{array}{c} \text{p}\downarrow \\ \text{s} \\ \text{s} \\ \text{s} \end{array} \frac{S([! R, ?T], [R, T])}{S([! R, ?T], R, T)} \quad \begin{array}{c} \text{b}\downarrow \end{array} \frac{S([! R, R], ?T, T)}{S([! R, R], ?T)} .$$

Here, a new instance of $b\downarrow$ is introduced. After all $b\uparrow$ have reached the top of the derivation, the instances of $b\downarrow$ are permuted down by the dual procedure, where new instances of $b\uparrow$ might be introduced; and so on.

The problem is to show that this process, shown in Fig. 4, does terminate eventually, which is done in two steps. First, the assumption of a non-termination is reduced to the existence of a derivation

$$\Delta \left[\{s, q\downarrow, q\uparrow\} \right], \left[(!R_1, ?T_1), (!R_2, ?T_2), \dots, (!R_n, ?T_n) \right] ,$$

for some $n \geq 1$ and structures $R_1, \dots, R_n, T_1, \dots, T_n$. Second, we show that such a derivation cannot exist. For the first step we mainly rely on the methods used in [19] for a case without seq. We need only a little more effort to deal with the unit. However, the non-existence of the derivation Δ is more difficult to prove for the system $\{s, q\downarrow, q\uparrow\}$ than in the case where the rules $q\downarrow$ and $q\uparrow$ are not present.

After separating absorption, we have to separate weakening. For the first decomposition, we use the same method used in [19], with the only difference that there are more cases to consider because the units 1 and $\perp$ are collapsed. In the second decomposition, weakening is separated in the same way as absorption, but termination is much easier to show.

In the end, the interaction rules are separated from the core, which is similar to the separation of weakening in the first decomposition result.

4 Cut Elimination

The classical arguments for proving cut elimination in the sequent calculus rely on the following property: when the principal formulae in a cut are active in both branches, they determine which rules are applied immediately above the cut. This is a consequence of the fact that formulae have a root connective, and logical rules only hinge on that, and nowhere else in the formula.

This property does not hold in the calculus of structures. Further, since rules can be applied anywhere deep inside structures, everything can happen above a cut. This complicates considerably the task of proving cut elimination. On the other hand, a great simplification is made possible in the calculus of structures by the reduction of cut to its atomic form. The remaining difficulty is actually understanding what happens, while going up in a proof, *around* the atoms produced by an atomic cut. The two atoms of an atomic cut can be produced inside any structure, and they do not belong to distinct branches, as in the sequent calculus: complex interactions with their context are possible. As a consequence, our techniques are largely different than the traditional ones.

Two approaches to cut elimination in the calculus of structures have been explored in previous papers: in [10] we relied on permutations of rules, in [4] the authors relied on semantics. In this paper we use a third technique, called *splitting*, which has the advantage of being more uniform than the one based on permutations and which yields a much simpler case analysis. It also establishes a deep connection to the sequent calculus, at least for the fragments of systems that allow for a sequent calculus presentation (in this case, the commutative fragment). Since many systems are expressed in the sequent calculus, our method appears to be entirely general; still it is independent of the sequent calculus and of a complete semantics.

Splitting can be best understood by considering a sequent system with no weakening and absorption (or contraction). Consider for example multiplicative linear logic: If we have a proof of the sequent $\{\vdash F\{A \otimes B\}, \Gamma\}$, where $F\{A \otimes B\}$ is a formula that contains the subformula $A \otimes B$, we know for sure that somewhere in the proof there is one and only one instance of the $\otimes$ rule, which splits A and B along with their context. We are in the following situation:

$$\begin{array}{ccc}
 \begin{array}{c}
 \begin{array}{c} \triangleleft \Pi_1 \triangleright \end{array} & \begin{array}{c} \triangleleft \Pi_2 \triangleright \end{array} \\
 \frac{\vdash A, \Phi \quad \vdash B, \Psi}{\vdash A \otimes B, \Phi, \Psi} \otimes \\
 \hline
 \begin{array}{c} \Delta \\ \vdash F\{A \otimes B\}, \Gamma \end{array}
 \end{array}
 & \text{corresponds to} &
 \begin{array}{c}
 \Pi_2 \top \\
 [B, \Psi] \\
 \Pi_1 \parallel \\
 \frac{([A, \Phi], [B, \Psi])}{s} \\
 \frac{([A, \Phi], B), \Psi}{s} \\
 \frac{[(A, B), \Phi, \Psi]}{s} \\
 \Delta \parallel \\
 [F(A, B), \Gamma]
 \end{array}
 \end{array}$$

We can consider, as shown at the left, the proof for the given sequent as composed of three pieces, Δ , Π_1 and Π_2 . In the calculus of structures, many different proofs correspond to the sequent calculus one: they differ for the different possible

sequencing of rules, and because rules in the calculus of structures have smaller granularity and larger applicability. But, among all these proofs, there must also be one that fits the scheme at the right of the figure above. This precisely illustrates the idea behind the splitting technique.

The derivation Δ above implements a *context reduction* and a proper splitting. We can state, in general, these principles as follows:

- 1 Context reduction: If $S\{R\}$ is provable, then $S\{ \}$ can be reduced to the structure $[\{ \}, U]$, such that $[R, U]$ is provable. In the example above, $[F\{ \}, \Gamma]$ is reduced to $[\{ \}, \Gamma']$, for some Γ' . (Technically, without loss of generality, we have to prefix the whole proof by a $!$.)
- 2 Splitting: If $[(R, T), P]$ is provable, then P can be reduced to $[P_1, P_2]$, such that $[R, P_1]$ and $[T, P_2]$ are provable. In the example above Γ' is reduced to $[\Phi, \Psi]$. Splitting holds for all logical operators.

Context reduction is in turn proved by splitting, which is then at the core of the matter. The biggest difficulty resides in proving splitting, and this mainly requires finding the right induction measure. The splitting theorems follow, all gathered together in Theorem 4.2.

4.1 Definition We call NEL_m the system $NEL \setminus \{w\downarrow, b\downarrow\} = \{\circ\downarrow, a\downarrow, s, q\downarrow, p\downarrow\}$.

4.2 Theorem For all structures R, T and P : $\langle P_1; P_2 \rangle$

- 1 if $[\langle R; T \rangle, P]$ is provable in NEL_m then there exist P_1, P_2 and $\|_{NEL_m}$
such that $[R, P_1]$ and $[T, P_2]$ are provable in NEL_m ; P
- 2 if $[(R, T), P]$ is provable in NEL_m then there exist P_1, P_2 and $[P_1, P_2]$
such that $[R, P_1]$ and $[T, P_2]$ are provable in NEL_m ; $\|_{NEL_m}$
 P
- 3 if $[!R, P]$ is provable in NEL_m then there exist $P_1, \dots, P_h$, for $h > 0$, and $!P_1$
 $[?P_1, \dots, ?P_h]$. $\|_{NEL_m}$
such that $[R, P_1, \dots, P_h]$ is provable in NEL_m ; P
- 4 if $[?R, P]$ is provable in NEL_m then there exist P_1 and $\|_{NEL_m}$ such that P
 $[R, P_1]$ is provable in NEL_m .

As a notable consequence of splitting we have:

4.3 Theorem If $[a, P]$ is provable in NEL_m then there exists $\bar{a}$
 $\bar{a}$ such that $[\bar{a}, P]$ is provable in NEL_m . $\|_{NEL_m}$
 P

4.4 Theorem A structure R is provable in NEL_m iff $!R$ is provable in NEL_m .

The context reduction theorem follows. The previous theorem insures its general applicability, despite the $!$ prefixing:

4.5 Theorem For all R and $S\{ \}$ such that $S\{R\}$ is provable in NEL_m , there $!S\{X\}$
exists a structure U such that for all X there exist $!X, U]$ $\|_{NEL_m}$ and $\|_{NEL_m}$
 $[R, U]$

And now we can sketch the cut elimination argument: Given a proof, the second decomposition theorem moves absorption and weakening down; coabsorption and coweakening go to the top. But coabsorptions and coweakenings at the top of a proof can only be trivial, and disappear. We perform cut elimination, on a proof thus rearranged, by going down the proof and eliminating all instances of

$a\uparrow$, $q\uparrow$ and $p\uparrow$, as they are encountered. This can be verified rather easily by the reader, through a combined use of context reduction and splitting. For example, by using the theorems above one can remove a topmost instance of $a\uparrow$ in a proof by the following transformation:

$$\begin{array}{c} \text{ai} \uparrow \frac{\prod_{\text{NELm}} S(a, \bar{a})}{S\{\circ\}} \end{array} \rightarrow \begin{array}{c} \begin{array}{c} \text{oi} \downarrow \frac{\quad}{\circ} \\ \text{ai} \downarrow \frac{\quad}{![\bar{a}, a]} \end{array} \\ \begin{array}{c} \prod_{\text{NELm}} \\ ! [P_1, P_2] \\ \prod_{\text{NELm}} \\ ! [\circ, U] \\ \prod_{\text{NELm}} \\ ! S\{\circ\} \end{array} \end{array} \rightarrow \begin{array}{c} \prod_{\text{NELm}} \\ S\{\circ\} \end{array}$$

We apply context reduction to $S(\bar{a}, \bar{a})$, which gives us U such that $[(\bar{a}, \bar{a}), U]$ is provable in NELm. By splitting, U is then further reduced into $[P_1, P_2]$ such that $[a, P_1]$ and $[\bar{a}, P_2]$ are provable in NELm. By applying Theorem 4.3 twice, we obtain a proof of $\vdash S\{\circ\}$. By Theorem 4.4, there is then a proof of $S\{\circ\}$ in NELm, which can be plugged into the original proof, so that the topmost $\text{ai}\uparrow$ has disappeared.

The rules $\mathbf{q}\uparrow$ and $\mathbf{p}\uparrow$ can be shown admissible the same way: we reduce the context of a rule instance as dictated by the shape of the structures involved in the rewriting. The pieces so produced are then rearranged—this is conceptually similar to what happens in the sequent calculus.

This technique shows how admissibility can be proved uniformly, both for cut rules (the atomic ones) and the other up rules, which are actually very different rules than cut. So, our technique is much more general than cut elimination in the sequent calculus, for two reasons:

- 1 it applies to operators that admit no sequent calculus definition, as seq;
2 it can be used to show admissibility of non-infinitary rules that involve no
negation, like $\mathbf{q}\uparrow$ and $\mathbf{p}\uparrow$.

5 Conclusions and Future Work

We have shown a class of logical systems, built around system NEL, that integrate multiplicative commutativity and non-commutativity, together with exponentials. This has been done in the formalism of the calculus of structures, which allows us to obtain very simple systems. In addition, we get properties of locality, atomicity and modularity that do not hold in other known calculi.

System NEL was originally inspired by Retoré’s pomset logic [16]. There is research in progress to show that the multiplicative fragments of his logic and ours coincide. In this case, our system and the work [21] would explain why sequentialising pomset logic has been so hard and unfruitful. It should be possible to extend our system NEL to other logical operators, perhaps to full linear logic, and also to the self-dual modality associated to Retoré’s non-commutative operator [15]. In this paper we limited ourselves to the bare necessary to include MELL.

In a forthcoming paper we will show that NEL is Turing-complete. This result establishes an interesting boundary to MELL, whose decidability is still an open problem. If it turns out, as many believe, that MELL is decidable, then the boundary with undecidability is crossed by our simple extension to seq. This would give a precise technical content to the perceived difficulty of getting Turing-equivalence for MELL, namely the trouble in realising the tape of a Turing machine. In this sense, our sequentiality would be even more strongly motivated by a basic computational mechanism.

One of the biggest open problems we have is understanding when and why decomposition theorems work. They seem to have a strong relation to the notion of core system, but we fail to understand the deep reasons for this. For the time being we observe that decomposition theorems hold for all logics we studied so far (classical, linear and several commutative/non-commutative systems).

The calculus of structures generalises the sequent calculus for one-sided sequent systems, which correspond to logics with involutive negation. Preliminary work shows that it is also possible to design intuitionistic systems in the calculus of structures, by way of polarities. This promises to be an active area of research.

Proving cut elimination is more difficult than in the sequent calculus. On the other hand, the methods we used are more general than the traditional ones, and, we believe, unveil some fundamental properties of logical systems that were previously hidden. We make an essential use of a top-down symmetric notion of derivation, which leads to a reduction of the cut rule into constituents which are dual to the common logical rules.

We did not attempt to base our calculus on philosophical grounds. We believe that this can only happen after several systems are thoroughly studied and discussed. For the time being we are still collecting empirical evidence.

References

- [1] Samson Abramsky and Radha Jagadeesan. Games and full completeness for multiplicative linear logic. *Journal of Symbolic Logic*, 59(2):543–574, June 1994.
- [2] V. Michele Abrusci. Phase semantics and sequent calculus for pure noncommutative classical linear propositional logic. *Journal of Symbolic Logic*, 56(4):1403–1451, 1991.
- [3] V. Michele Abrusci and Paul Ruet. Non-commutative logic I: The multiplicative fragment. *Annals of Pure and Applied Logic*, 101(1):29–64, 2000.
- [4] Kai Brännler and Alwen Fernanto Tiu. A local system for classical logic. In R. Nieuwenhuis and A. Voronkov, editors, *LPAR 2001*, volume 2250 of *Lecture Notes in Artificial Intelligence*, pages 347–361. Springer-Verlag, 2001. On the web at: <http://www.ki.inf.tu-dresden.de/~kai/LocalClassicalLogic-lpar.pdf>.
- [5] Paola Bruscoli. A purely logical account of sequentiality in proof search. In Peter J. Stuckey, editor, *Logic Programming, 18th International Conference*, volume 2401 of *Lecture Notes in Artificial Intelligence*, pages 302–316. Springer-Verlag, 2002. On the web at: <http://www.ki.inf.tu-dresden.de/~paola/bvl/bvl.pdf>.

- [6] Arnaud Fleury and Christian Retoré. The mix rule. *Mathematical Structures in Computer Science*, 4(2):273–285, 1994.
- [7] Gerhard Gentzen. Investigations into logical deduction. In M. E. Szabo, editor, *The Collected Papers of Gerhard Gentzen*, pages 68–131. North-Holland, Amsterdam, 1969.
- [8] Jean-Yves Girard. Linear logic. *Theoretical Computer Science*, 50:1–102, 1987.
- [9] Alessio Guglielmi. A calculus of order and interaction. Technical Report WV-99-04, Dresden University of Technology, 1999. On the web at: <http://www.ki.inf.tu-dresden.de/~guglielm/Research/Gug/Gug.pdf>.
- [10] Alessio Guglielmi and Lutz Straßburger. Non-commutativity and MELL in the calculus of structures. In L. Fribourg, editor, *CSL 2001*, volume 2142 of *Lecture Notes in Computer Science*, pages 54–68. Springer-Verlag, 2001. On the web at: <http://www.ki.inf.tu-dresden.de/~guglielm/Research/GugStra/GugStra.pdf>.
- [11] Alessio Guglielmi and Lutz Straßburger. A non-commutative extension of MELL. Technical Report WV-02-03, Dresden University of Technology, 2002. On the web at: <http://www.ki.inf.tu-dresden.de/~guglielm/Research/NEL/NELbig.pdf>.
- [12] Joachim Lambek. The mathematics of sentence structure. *American Mathematical Monthly*, 65:154–169, 1958.
- [13] P. Lincoln, J. Mitchell, A. Scedrov, and N. Shankar. Decision problems for propositional linear logic. *Annals of Pure and Applied Logic*, 56(1–3):239–311, 1992.
- [14] Christian Retoré. *Réseaux et Séquents Ordonnés*. Thèse de Doctorat, spécialité mathématiques, Université Paris 7, February 1993.
- [15] Christian Retoré. A self-dual modality for “Before” in the category of coherence spaces and in the category of hypercoherences. Technical Report 2432, INRIA, 1994.
- [16] Christian Retoré. Pomset logic: A non-commutative extension of classical linear logic. In Ph. de Groote and J. R. Hindley, editors, *TLCA’97*, volume 1210 of *Lecture Notes in Computer Science*, pages 300–318, 1997.
- [17] Christian Retoré. Pomset logic as a calculus of directed cographs. In V. M. Abrusci and C. Casadio, editors, *Dynamic Perspectives in Logic and Linguistics*, pages 221–247. Bulzoni, Roma, 1999. Also available as INRIA Rapport de Recherche RR-3714.
- [18] Paul Ruet. Non-commutative logic II: Sequent calculus and phase semantics. *Mathematical Structures in Computer Science*, 10:277–312, 2000.
- [19] Lutz Straßburger. MELL in the calculus of structures. Technical Report WV-01-03, Dresden University of Technology, 2001. On the web at: <http://www.ki.inf.tu-dresden.de/~lutz/els.pdf>, submitted to TCS.
- [20] Lutz Straßburger. A local system for linear logic. Technical Report WV-02-01, Dresden University of Technology, 2002. Accepted at LPAR’02, this volume. On the web at: <http://www.ki.inf.tu-dresden.de/~lutz/lis-lpar.pdf>.
- [21] Alwen Fernanto Tiu. Properties of a logical system in the calculus of structures. Technical Report WV-01-06, Dresden University of Technology, 2001. On the web at: <http://www.cse.psu.edu/~tiu/thesisc.pdf>.
- [22] David N. Yetter. Quantales and (noncommutative) linear logic. *Journal of Symbolic Logic*, 55(1):41–64, 1990.

A System of Interaction and Structure

Alessio Guglielmi

Technische Universität Dresden

Fakultät Informatik - 01062 Dresden - Germany

Alessio.Guglielmi@Inf.TU-Dresden.DE

Abstract *This paper introduces a logical system, called BV, which extends multiplicative linear logic by a non-commutative self-dual logical operator. This extension is particularly challenging for the sequent calculus, and so far it is not achieved therein. It becomes very natural in a new formalism, called the calculus of structures, which is the main contribution of this work. Structures are formulae subject to certain equational laws typical of sequents. The calculus of structures is obtained by generalising the sequent calculus in such a way that a new top-down symmetry of derivations is observed, and it employs inference rules that rewrite inside structures at any depth. These properties, in addition to allowing the design of BV, yield a modular proof of cut elimination.*

Table of Contents

1	Introduction	1
2	Structures and Relation Webs	5
1	Structures	5
2	Relation Webs	10
3	Synthesis of a Formal System	17
1	Merge Rules and Relation Webs	20
2	The Structure Fragment	24
3	Comments on the Structure Fragment	30
4	Interaction	31
4	Breaking the Symmetry	36
1	The Splitting Theorem	37
2	Admissibility of the Up Fragment	46
5	Relation with Linear Logic	51
6	Conclusions	56
	Acknowledgments	60
	References	60
	Index of Symbols	62

1 Introduction

In the beginning of this enterprise I only wanted to define a deductive system mixing commutative and non-commutative logical operators. The specific variant I was looking

for is a conservative extension of multiplicative linear logic plus mix, obtained by adding a self-dual non-commutative connective. This problem has been studied mainly by Retoré [19, 20]: he has proof nets for his logic, which is called *pomset logic*, and cut elimination therein, but, despite many efforts, nobody has been able so far to provide a sequent system for that logic. My challenge was to bring pomset logic into the realm of the sequent calculus, for two reasons: 1) It is interesting to see *why* such a simple logic requires so much effort to be expressed in the most universal proof-theoretical formalism. 2) If one wants to extend pomset logic to more expressive logics, then the sequent calculus usually is a better formalism than proof nets, because it is more versatile, for example with exponentials and additives.

What my colleagues and I found in this research has been surprising: there are deep reasons for this kind of logic not to be expressible in the sequent calculus, and there is a simple formalism, which we call the *calculus of structures*, that is instead able to express self-dual non-commutativity with great ease. In fact, self-dual non-commutative operators naturally generate a class of provable formulae whose proofs depend on accessing subformulae at arbitrary depths, what makes for an impossible challenge to the sequent calculus. The new formalism is more general than the sequent calculus for logics with involutive negation, like classical and linear logic, and allows a much more refined proof theory than possible in the sequent calculus, without sacrificing simplicity. In particular, rules in the calculus of structures can access arbitrarily nested subformulae.

This paper represents the beginning of our research program in the calculus of structures, and focuses on designing a deductive system with the characteristics mentioned above for pomset logic. It is still open whether the logic in this paper, called BV, is the same as pomset logic. We conjecture that it is actually the same logic, but one crucial step is still missing, at the time of this writing, in the equivalence proof. This paper is the first in a planned series of three papers dedicated to BV. In the present, first part, I obtain BV from a certain semantic idea, and I show cut elimination for it. In the second part, Alwen Tiu will show why BV cannot be defined in any sequent system [25]. In the third part, some of my colleagues will hopefully show the equivalence of BV and pomset logic, this way explaining why it was impossible to find a sequent system for pomset logic. Some of the results in this paper have been already shown by Straßburger and myself in [9, 10] and used by Bruscoli in [7].

Pomset logic naturally derives from the study of coherence spaces for multiplicative linear logic (see Girard's [8]), and its self-dual operator has a close correspondence to sequential operators as defined, for example, in process algebras. The cut elimination procedure in the proof nets of pomset logic gets sequentialised by the non-commutative links; this naturally induces a computational model where sequentiality plays a role as important as parallelism, which is interesting in the light of the Curry-Howard correspondence. Non-commutative logics are also important in linguistics, their use dating back to the Lambek calculus [12]. As well as from its semantic origins, BV can be motivated by the same reasons as for pomset logic: essentially, non-commutativity brings sequentiality, and sequentiality is an archetypal algebraic composition both for human and computer languages. Self-dual non-commutativity captures very precisely the sequentiality notion of many process algebras, CCS [14] in the first place, as Bruscoli shows in [7]. So, at a very

concrete level, this paper is about a simple logic that can help understand several linguistic phenomena. For it, I provide a simple deductive system and prove cut elimination.

On the other hand, at a different level, this paper only uses BV as an experiment for tuning up a workbench where other ambitious proof-theoretical problems could be studied. I present here a new formalism, the calculus of structures, and some of the proof-theoretical methods associated to it. The calculus of structures is a generalisation of the one-sided sequent calculus, which is also called the *Gentzen-Schütte* calculus, (see, for example, [26]). Many logics with involutive negation and De Morgan laws can be defined in the one-sided sequent calculus: in that case, translating them into the calculus of structures is a trivial (and also uninteresting) operation. What makes the calculus of structures interesting is the possibility of defining logics by employing concepts fundamentally different from those in the sequent calculus. We can isolate two ideas:

- 1 *Deep inference*: inference rules in the calculus of structures operate anywhere inside expressions, they are not confined to the outermost subformulae around the roots of formula trees.
- 2 *Top-down symmetry*: contrary to the sequent calculus, natural deduction and other formalisms where derivations are essentially based on trees, in our new formalism derivations can be flipped upside-down and negated, and they remain valid (this corresponds to $A \rightarrow B$ being equivalent to $\bar{B} \rightarrow \bar{A}$).

These ideas find a simple technical realisation by employing the unifying notion of ‘structure’.

I borrowed the terminology from Belnap’s [16], and in general from the tradition in philosophical and substructural logics [18]: a *structure* is an expression intermediate between a (one-sided) sequent and a formula. More precisely, it is an ordinary logical formula modulo an equational theory of the kind typically imposed on sequents. From a practical viewpoint, logical connectives disappear (in particular connectives at the root of formula trees), and all logical rules become *structural* (as opposed to *logical*), in the sense that they deal with the relative position of substructures in a structure. Structures are the only kind of expression allowed, and inference rules are simply rewriting rules on structures, whence the name ‘calculus of structures’. Of course, the freedom allowed by this formalism is dangerous. One could use it in a wild way, and lose every proof-theoretical property. I will be very cautious in defining rules, and actually the main part of our research is understanding and defining the methodologies necessary for making an educated and profitable use of the new freedom.

The main point to note is that in the calculus of structures it is possible to do proof theory: we can define a cut rule, and cut elimination makes sense, just as in the sequent calculus. Moreover, there is an analogue to the subformula property, which guarantees that any given rule (except for the cut and perhaps other special rules) has finite applicability. These features make the calculus of structures closer to the sequent calculus than to any other formalism. On the other hand, cut elimination in the calculus of structures is completely different from that in the sequent calculus, at least when systems are designed by using the deep inference feature. In fact, in the sequent calculus the cut elimination procedure depends crucially on the existence of a root connective, and this is not the

case in the calculus of structures. The complication comes essentially from deep inference; luckily, the top-down symmetry contributes to simplifying the situation considerably. As a consequence of the symmetry, in fact, we have that the cut rule can always be reduced trivially to its atomic version, i.e., the eigenformulae are atoms. This is the perfect dual of the typical sequent calculus fact that generic identity rules can be replaced by their atomic counterparts.

In this paper I introduce one of the two existing techniques for proving cut elimination in the calculus of structures, and I call it *splitting* (the other one is called *decomposition*, see [9, 10], and it is best used in conjunction with splitting). An important difference of the calculus of structures with respect to the sequent calculus is that the cut rule can be equivalently divided into several rules. Often, only one of these rules is infinitary, the atomic cut, but all the rules in the decomposition of cut can be shown admissible. An advantage is that one can have a wide range of equivalent systems (one for each combination of admissible rules), with varying degrees of expressiveness in proof construction. All the admissible rules are in fact independent, and their admissibility can be shown independently by way of splitting. For big logical systems, like linear logic, one can easily get tens of thousands of equivalent systems without much effort [23]. This *modularity* is ultimately made possible by the top-down symmetry, and modularity is of course important both conceptually and for the typical computer science applications.

The ideas of structure, deep inference and top-down symmetry, all come from a certain combinatorial, graph-like representation of logical formulae that I call *relation webs*. Relation webs are used in this paper as both a semantics and a very abstract, distributed computational model for BV. They make for a compact representation of formulae that: 1) independently justifies the equational theory imposed on structures; 2) induces an extremely natural notion of rewriting, which actually generates BV independently of coherence spaces or any other formalism. Relation webs play a crucial role in Tiu’s proof of inadequacy of the sequent calculus for system BV [25], where deep inference is shown to be an essential feature for representing BV—a feature that the sequent calculus does not possess. I will argue about relation webs having a broad applicability. In fact, a certain characterisation theorem for relation webs, which is crucial in our treatment, scales up to the generic case of a logic made by any number of different multiplicative logical relations.

Relation webs justify the *structure fragment* of BV, i.e., the rules responsible for the combinatorial behaviour of the logic, independently of negation. The *interaction fragment* is made by the rules corresponding to identity and cut in the sequent calculus, i.e., the rules which depend on negation. The latter fragment also enjoys a nice interpretation in relation webs, but, at least at present, the integration of interaction and structure fragments is best studied in the calculus of structures, owing to the relatively poor development of relation webs so far. I decided to underscore the importance of interaction and structure in the paper’s title because I have reasons to believe that they are actually a key to understand in a new way many proof-theoretical phenomena. I will not give any hard evidence of this belief in this paper, because we are still in a phase in which we have to collect evidence from the study of several deductive systems before attempting a more philosophical explanation of our observations. On the other hand, we studied already a number of systems and they all fit in the interaction/structure pattern outlined here; I invite the reader to form a

personal opinion on this matter by reading the papers on classical [6] and linear logic [24, 23], as well as the extensions of system BV [10].

In summary, these are the contributions of this paper:

- 1 A new proof-theoretical formalism, the calculus of structures, which allows to express new logical systems and has proof-theoretical properties of special interest for computer science; the formalism comes with new, non-trivial techniques for proving proof-theoretical results.
- 2 A deductive system, called BV, for a logic which extends multiplicative linear logic with a logical operator whose interpretation is a broad notion of sequentality.
- 3 A combinatorial representation of formulae, called relation web, which allows an independent, semantic justification for BV.

Section 2 introduces structures together with relation webs and an informal representation for structures that should help understanding them intuitively. Section 3 introduces the calculus of structures and derives system BV together with its cut rule starting from relation webs. System BV and cut elimination are discussed in Section 4. Finally, in Section 5, I will show how BV is a conservative extension of linear logic plus mix.

2 Structures and Relation Webs

The notion of structure is not complicated, but it is useful to have an intuitive idea of how it relates to traditional proof-theoretical notions. Very roughly, a structure is both a formula and a sequent, but, as we will see soon, it also captures aspects of proof nets. A good way to understand structures, at least for what matters to this paper, is through their graph representation. The first subsection is dedicated to structures and their representation, the second one to relation webs.

2.1 Structures

Since structures also capture some aspects of proofs, it should be appropriate to start introducing them by an example from proof construction in multiplicative linear logic, whose management by structures is also of independent interest. In a one-sided sequent presentation, the multiplicative conjunction $\otimes$ (*times*) of linear logic is defined by:

$$\otimes \frac{\vdash A, \Phi \quad \vdash B, \Psi}{\vdash A \otimes B, \Phi, \Psi} .$$

Instances of $\otimes$ can be seen as elementary steps in a computation that corresponds to the bottom-up building of a proof. This perspective is called the *proof search as computation*, or *proof construction*, paradigm. From this point of view the rule $\otimes$ above has a serious flaw: when the rule is applied we have to decide how to split the context of the main formula, and if n formulae are in the multiset Φ, Ψ , there are 2^n ways to assign them to Φ and Ψ . This is an exponential source of probably unwanted non-determinism.

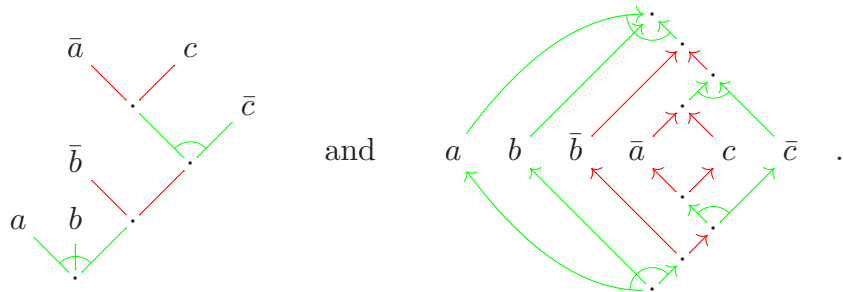
Of course, we could adopt an approach external to the sequent calculus, an *implementation* that controls the use of the rule *a posteriori*: no commitment is made at the time the rule is applied, and resources in the context are used based on necessity along the search for the proof, ideally updating the actual instance of the rule $\otimes$ until a valid proof is formed. This ‘lazy’ approach is similar to the use of most general unifiers when having to decide how to instantiate quantified variables. It is of course a very reasonable solution, already adopted by Hodas and Miller in the case of intuitionistic linear logic [11].

There still remains the curiosity of whether this problem could be solved *inside* a deductive system. We should create clusters of formulae, corresponding to contexts to be split, whence each formula could be picked up on demand. In the traditional sequent calculus this solution is not achievable by simple means. What really counts, while building a proof in the bottom-up way in the sequent calculus, is the frontier of hypotheses, whose natural structure is simply a *multiset*. Instead, we need to keep a *tree*, whose structure takes care of the nesting of clusters. Consider for example the following proof in multiplicative linear logic, where $\bar{a}$ is the negation of the atom a and the connective $\wp$ is called *par*:

$$\begin{array}{c}
 \text{id} \frac{}{\vdash a, \bar{a}} \quad \text{id} \frac{}{\vdash c, \bar{c}} \\
 \otimes \frac{}{\vdash a, \bar{a} \otimes c, \bar{c}} \\
 \wp \frac{}{\vdash a, (\bar{a} \otimes c) \wp \bar{c}} \\
 \otimes \frac{}{\vdash a, b, \bar{b} \otimes ((\bar{a} \otimes c) \wp \bar{c})} \\
 \wp \frac{}{\vdash a, b \wp (\bar{b} \otimes ((\bar{a} \otimes c) \wp \bar{c}))} \\
 \wp \frac{}{\vdash a \wp (b \wp (\bar{b} \otimes ((\bar{a} \otimes c) \wp \bar{c})))} .
 \end{array}$$

In this case the appropriate partitions of contexts have been found, and we got a proof.

Consider now the following two graphs:



The tree-like structure at the left is a representation of the conclusion of the proof above. Times-connected formulae are linked to the roots by simple arcs; these formulae are in a relation that I call *copar* (instead of times). Par-connected formulae are connected to the roots by arcs that are in a special relation, represented by the bow. The only purpose of the bows is to distinguish pars from copars; I could have equivalently put bows on copar arcs. The graph at the left is more or less a proof net without the axiom links, but we can also consider it as a partial representation of the proof above, which only deals with the $\wp$ rule instances. This example is important and I invite the reader to get back to it

while reading this paper. The deductive mechanism we are going to see will take care of the $\otimes$ rule instances which are present in the multiplicative linear logic proof and absent from the graphs above.

The graph at the right is obtained from the left one by taking the horizontal mirror image of the lower part above the row of atoms and by directing arcs. Nodes that are not atoms have the only meaning of keeping things together. Being redundant, this kind of structure is probably silly to use with linear logic; and anyway, my main interest here is not in the partitioning of the times context. I will slightly generalise these structures in order to include a new, non-commutative logical relation called *seq*, for which the top-down symmetry will not be trivial any more.

Formally to deal with these graphs, I will now introduce a bit of syntax, alternative to the one of linear logic. In this language the formula to be proved above can be written $[a, b, (\bar{b}, [(\bar{a}, c), \bar{c}])]$ and expressions like this are called *structures*. Instead of using binary connectives for defining logical relations, I consider relations induced by a context, meaning that, for example, $[R_1, \dots, R_h]$ is a par structure, where the structures inside it are to be considered connected by pars. Since par is commutative and associative (and the commutativity and associativity equivalences are decidable), I will not distinguish structures based on ordering or on the nesting of a par structure into another par. The same is true for copar: a copar structure $(R_1, \dots, R_h)$ has the same properties as a par one.

2.1.1 Definition There are infinitely many *positive atoms* and infinitely many *negative atoms*. Atoms, no matter whether positive or negative, are denoted by $a, b, c, \dots$. *Structures*, denoted by S, P, Q, R, T, U, V and X , are generated by

$$S ::= \circ \mid a \mid \underbrace{\langle S; \dots; S \rangle}_{>0} \mid \underbrace{[S, \dots, S]}_{>0} \mid \underbrace{(S, \dots, S)}_{>0} \mid \bar{S} \quad ,$$

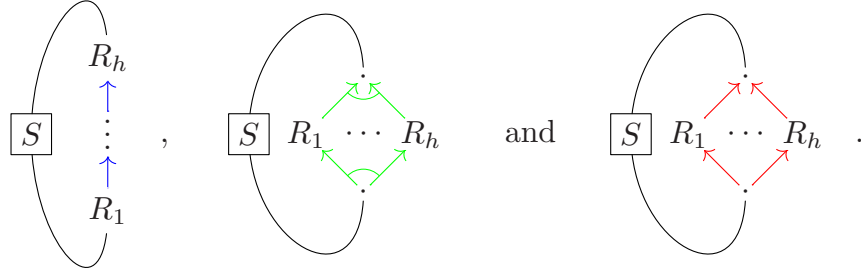
where $\circ$, the *unit*, is not an atom. $\langle S_1; \dots; S_h \rangle$ is called a *seq structure*, $[S_1, \dots, S_h]$ is called a *par structure* and $(S_1, \dots, S_h)$ is called a *copar structure*; $\bar{S}$ is the *negation* of the structure S ; a negated atom $\bar{a}$ is a negative atom if a is positive and a positive one if a is negative. We can designate a special atom as a *hole*, denoted by $\{ \}$, whose purpose is to indicate a specific place in a given structure, where sometimes other structures are plugged in; structures with a hole that does not appear in the scope of a negation are denoted as in $S\{ \}$, and are called *structure contexts*. The structure R is a *substructure* of $S\{R\}$, and $S\{ \}$ is the *context* of R therein.

2.1.2 Notation When structural brackets of any kind surround the content of a hole, hole braces will be omitted. For example, $S[a, b]$ stands for $S\{[a, b]\}$.

2.1.3 Notation A letter implicitly establishes the class to which an object belongs: for example, when we write S , we denote a structure, without necessarily saying explicitly that S is a structure.

The structures $S\langle R_1; \dots; R_h \rangle$, $S[R_1, \dots, R_h]$ and $S(R_1, \dots, R_h)$ are respectively rep-

resented as



Arrows between non-seq structures in a seq structure context will be omitted.

Associativity holds for seq structures, but commutativity does not. Contrarily to their commutative counterparts, which live in what we can consider the horizontal line of space, the seq structures can make use of time. For example, consider the three structures in Figure 8; they correspond, from left to right, to the expressions

$$[\langle [P, U]; (Q, R) \rangle, V] \quad , \quad \langle P; (Q, R) \rangle, U, V] \quad \text{and} \quad [P; \langle (Q, U), R \rangle, V] \quad .$$

An intuitive understanding is the following: Nodes represent points in space-time, and the order induced by arcs can be mapped consistently into a total order, which we can consider time. The structures in a par can interact at all times through the span of the par arc that connects them to the rest of the structure. The structures in a copar, instead, cannot interact. I could have also considered the ‘cointerpretation’ in which the structures in a par cannot interact and the structures in a copar can. No matter the interpretation, the structures in a seq should not be allowed to interact, because they stay in different time spans. Figure 8 shows two possible rewritings of the central structure, which respect our space-temporal intuition. I will get back to that figure later on.

Let us now turn our attention to negation. Negation is involutive and obeys the usual De Morgan laws for par and copar: it switches them. It is natural to state that $\langle S_1; \dots; S_h \rangle = \langle \bar{S}_1; \dots; \bar{S}_h \rangle$: time is not reversed through negation. The behaviour of negation with respect to seq is in accord with the similar case in Retoré’s pomset logic [20], where a commutative/non-commutative linear logic is derived from coherence semantics. By applying these laws, negation can always be pushed inward to atoms.

I will equip structures with a decidable equivalence relation that takes care of associativity, commutativity and negation. This places the burden of realising these properties on our pattern-matching abilities instead of on structural rules of the system. The same happens in sequent systems where the exchange rule becomes commutativity in sequents, or the contraction rule becomes idempotency, etc. All the three kinds of structures enjoy associativity, but only par and copar are commutative.

2.1.4 Definition Structures are considered equivalent modulo the relation $\equiv$, which is the minimal equivalence relation defined by the axioms in Figure 1. There, $\vec{R}$, $\vec{T}$ and $\vec{U}$ stand for finite, non-empty sequences of structures. A structure, or a structure context, is said to be *in normal form* when the only negated structures appearing in it are atoms, no unit $\circ$ appears in it and no brackets can be eliminated while maintaining equivalence. If structures R and T are such that $R \neq \circ \neq T$, then the structure $\langle R; T \rangle$ is a *proper seq*, the structure $[R, T]$ is a *proper par* and the structure (R, T) is a *proper copar*. A structure

Associativity

$$\begin{aligned}\langle \vec{R}; \langle \vec{T}; \vec{U} \rangle &= \langle \vec{R}; \vec{T}; \vec{U} \rangle \\ [\vec{R}, [\vec{T}]] &= [\vec{R}, \vec{T}] \\ (\vec{R}, (\vec{T})) &= (\vec{R}, \vec{T})\end{aligned}$$

Commutativity

$$\begin{aligned}[\vec{R}, \vec{T}] &= [\vec{T}, \vec{R}] \\ (\vec{R}, \vec{T}) &= (\vec{T}, \vec{R})\end{aligned}$$

Negation**Singleton**

$$\langle R \rangle = [R] = (R) = R$$

Unit

$$\begin{aligned}\langle \circ; \vec{R} \rangle &= \langle \vec{R}; \circ \rangle = \langle \vec{R} \rangle \\ [\circ, \vec{R}] &= [\vec{R}] \\ (\circ, \vec{R}) &= (\vec{R})\end{aligned}$$

$$\begin{aligned}\bar{\circ} &= \circ \\ \overline{\langle R; T \rangle} &= \langle \bar{R}; \bar{T} \rangle \\ \overline{[R, T]} &= (\bar{R}, \bar{T}) \\ \overline{(R, T)} &= [\bar{R}, \bar{T}] \\ \bar{\bar{R}} &= R\end{aligned}$$

Contextual Closure

$$\text{if } R = T \text{ then } S\{R\} = S\{T\}$$

Fig. 1 *Syntactic equivalence =*

context $S\{ \}$ is a proper seq (a proper par, a proper copar) if, for all $X \neq \circ$, the structure $S\{X\}$ is a proper seq (a proper par, a proper copar). The structures whose normal forms do not contain seq structures are called *flat* (if one normal form does not contain seq, none does).

The set of equations in Figure 1 is not minimal, of course, but there is no real reason to minimise it. The structures $\overline{[a, \circ, b]}$, $(([\circ, b]), \langle \bar{a} \rangle)$ and $(\bar{a}, \bar{\circ}, \bar{b})$ are all equivalent by $=$, but they are not in normal form; $(\bar{a}, \bar{b})$ is equivalent to them and normal, as well as $(\bar{b}, \bar{a})$; so, all the previous structures are flat. Since structures are considered equivalent under $=$, the structure $[\circ, \langle a; b \rangle]$ is a proper seq, but not a proper par or a proper copar; $\langle a; [\circ, b] \rangle$ is a proper seq, and $\langle a; [\{ \}, b] \rangle$ is a proper seq context, while $[\{ \}, b]$ is a proper par context.

2.1.5 Remark All structures can equivalently be considered in normal form, since negations can always be pushed inward to atoms by using the negation axioms, and units can be removed, as well as extra brackets (by associativity and singleton laws). Every structure can only be equivalent either to the unit, or to an atom, or, mutually exclusively, to a proper seq, or a proper par, or a proper copar.

Please note that negation obeys De Morgan-like equivalences for par and copar; seq is self-dual and ordering is maintained through negation. The reader should not be alarmed by the equation $\bar{\circ} = \circ$. The use of the unit $\circ$ is different from the traditional use of logical constants, like t and f . In our case $\circ$ is just a convenient, syntactic mark that allows us to get a compact form for inference rules; I do not plan to interpret it semantically. There are anyway consistent formal systems where the units collapse into a unique one for all logical operators: for example, multiplicative linear logic with mix and nullary mix, a logic of which BV is a conservative extension (See Remark 5.9).

2.1.6 Definition Given a structure S , we talk about *atom occurrences* when considering all the atoms appearing in S as distinct (for example, by indexing them so that two atoms

which are equal get different indices); therefore, in $\langle a; a \rangle$ there are two atom occurrences, those of the atom a ; in the structure $\bar{a}$, there is only one atom occurrence, the negative atom $\bar{a}$. The notation $\text{occ } S$ indicates the set of all the atom occurrences appearing in S . The *size* of S is the cardinality of the set $\text{occ } S$. We will use the expression ‘atom occurrences’ also outside of structures, whenever we consider distinct possibly multiple appearances of the same atom.

The set $\text{occ } S$ could be defined as the multiset of the atoms of S , or of S in normal form. Note that $\text{occ } \circ = \emptyset$; also note that $\text{occ } \langle S; S' \rangle = \text{occ } [S, S'] = \text{occ } (S, S') = \text{occ } S \cup \text{occ } S'$ is true only if $\text{occ } S$ and $\text{occ } S'$ are disjoint, and we can always assume this without loss of generality.

2.2 Relation Webs

The informal graphic representation of structures seen above partly justifies the choice of equations we imposed on structures. It is a weak justification, which only relies on the representation being intuitively appealing. In this subsection I will offer a strong justification. We will see another representation of structures, this time formal: *relation webs*. Contrary to the ‘space-time’ representation, relation webs do not have an immediate correspondence to the inference rules we are going to see later, but they have other important features:

- 1 There is a unique relation web for every equivalence class of structures under $=$.
- 2 The formal system BV will be derived from relation webs by asking for a certain conservation property to hold while we manipulate relation webs.

Relation webs are used as a sort of semantics, which gives independent and faithful account of inference rules. I will develop them only in relation to the structure fragment of BV. This is enough to establish Tiu’s results on the inadequacy of the sequent calculus for BV in [25], where relation webs are used to show that certain structures are not provable.

The readers who wish only to understand the behaviour of system BV, possibly also in relation to the sequent calculus of linear logic, need not read this subsection. It is safe to ignore everything about relation webs and still read the rest of the paper, with the exception of Subsection 3.1.

Consider $[R_1, \dots, R_h]$: let us establish that for distinct i and j and for all the atoms a appearing in R_i and b appearing in R_j the relation $a \downarrow b$ holds (so $\downarrow$ is symmetric). Analogously, two distinct structures R and T in a copar induce on their atoms a relation $a \uparrow b$, where a is in R and b is in T . For example, in $[a, b, (\bar{b}, [(\bar{a}, c), \bar{c}])]$ these relations are determined: $a \downarrow b$, $a \downarrow \bar{b}$, $a \downarrow \bar{a}$, $a \downarrow c$, $a \downarrow \bar{c}$, $b \downarrow \bar{b}$, $b \downarrow \bar{a}$, $b \downarrow c$, $b \downarrow \bar{c}$, $\bar{b} \uparrow \bar{a}$, $\bar{b} \uparrow c$, $\bar{b} \uparrow \bar{c}$, $\bar{a} \uparrow c$, $\bar{a} \downarrow \bar{c}$, $c \downarrow \bar{c}$ (the symmetric relations have been omitted, for example $\bar{a} \uparrow \bar{b}$). Let us add that $\langle S_1; \dots; S_h \rangle$ induces the relation $a \triangleleft b$ for all the atoms a in S_i and b in S_j such that $1 \leq i < j \leq h$ (so $\triangleleft$ is not symmetric).

2.2.1 Definition Given a structure S in normal form, the four *structural relations* $\triangleleft_S$ (*seq*), $\triangleright_S$ (*coseq*), $\downarrow_S$ (*par*) and $\uparrow_S$ (*copar*) are the minimal sets such that $\triangleleft_S, \triangleright_S, \downarrow_S, \uparrow_S \subset (\text{occ } S)^2$ and, for every $S' \{ \}$, U and V and for every a in U and b in V , the following hold:

- 1 if $S = S' \langle U; V \rangle$ then $a \triangleleft_S b$ and $b \triangleright_S a$;

- 2** if $S = S'[U, V]$ then $a \downarrow_S b$;
- 3** if $S = S'(U, V)$ then $a \uparrow_S b$.

To a structure that is not in normal form we associate the structural relations obtained from any of its normal forms, since they yield the same. The quadruple $(\text{occ } S, \triangleleft_S, \downarrow_S, \uparrow_S)$ is called the *relation web* (or simply *web*) of S , written $w S$. We can abolish the subscripts in $\triangleleft_S, \triangleright_S, \downarrow_S$ and $\uparrow_S$ when they are not necessary. Given two sets of atom occurrences μ and ν , we write $\mu \triangleleft \nu$, $\mu \triangleright \nu$, $\mu \downarrow \nu$ and $\mu \uparrow \nu$ to indicate situations where, for every a in μ and for every b in ν , the following hold, respectively: $a \triangleleft b$, $a \triangleright b$, $a \downarrow b$ and $a \uparrow b$. We represent structural relations between occurrences of atoms by drawing $a \rightsquigarrow b$ when $a \triangleleft b$ (and $b \triangleright a$), $a \leftrightsquigarrow b$ when $a \triangleleft b$ or $a \triangleright b$, $a \text{---} b$ when $a \downarrow b$ and $a \text{---} b$ when $a \uparrow b$. Dashed arrows represent negations of structural relations.

For example, in $(\langle \bar{a}; b \rangle, \langle c, \bar{d} \rangle) = (\langle a; \bar{b} \rangle, [\bar{c}, d])$ these relations are determined: $a \triangleleft \bar{b}$, $a \uparrow \bar{c}$, $a \uparrow d$, $\bar{b} \triangleright a$, $\bar{b} \uparrow \bar{c}$, $\bar{b} \uparrow d$, $\bar{c} \uparrow a$, $\bar{c} \uparrow \bar{b}$, $\bar{c} \downarrow d$, $d \uparrow a$, $d \uparrow \bar{b}$, $d \downarrow \bar{c}$.

The graphical notation for structural relations will be extensively used in the following: please note that the graphs obtained from structural relations are not the same as the ‘space-time’ ones representing structures!

2.2.2 Remark A structure S such that $w S = (\text{occ } S, \triangleleft, \downarrow, \uparrow)$ is flat iff $\triangleleft = \emptyset$.

It is easy to see from the definitions that all the atoms in a substructure are in the same structural relation with respect to each of the atoms in the context surrounding the substructure:

2.2.3 Proposition *Given a structure $S\{R\}$ and two atom occurrences a in $S\{ \}$ and b in R , if $a \triangleleft b$ (respectively, $a \triangleright b$, $a \downarrow b$, $a \uparrow b$) then $a \triangleleft c$ (respectively, $a \triangleright c$, $a \downarrow c$, $a \uparrow c$) for all the atom occurrences c in R .*

The given syntax of structures, and the use of the equivalence $=$, help to focus the system more on meaning than on representation. A structure should picture, with the minimum amount of ink and ambiguity, a certain situation, where what really matters are atoms and their mutual structural relations: the web of a structure displays this information. However, taking a bunch of atoms and assigning structural relations to couples of them does not guarantee to produce a valid structure. Two questions are then in order: 1) when does an assignment of relations to atoms actually give a structure, and 2) whether the structures having a given web are equivalent by $=$, or not.

This subsection answers both questions above. At this point the reader in a hurry knows enough to jump immediately to the next section, after reading the statement of Theorem 2.2.9. What follows is a characterisation of structures in terms of seven properties of structural relations. The most intriguing (and less expected) of them is what I call the ‘square property’. It has been found with much help from a paper by Möhring, [15], where a simpler case about series-parallel orders is studied. The next theorem shows that certain properties are necessary for structures.

2.2.4 Theorem *Given S and its associated structural relations $\triangleleft, \triangleright, \downarrow$ and $\uparrow$, the following properties hold, where a, b, c and d are distinct atom occurrences in S :*

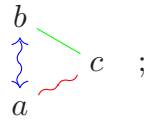
- s_1 None of $\triangleleft, \triangleright, \downarrow$ and $\uparrow$ is reflexive: $\neg(a \triangleleft a)$, $\neg(a \triangleright a)$, $\neg(a \downarrow a)$, $\neg(a \uparrow a)$.
- s_2 One and only one among $a \triangleleft b$, $a \triangleright b$, $a \downarrow b$ and $a \uparrow b$ holds.

- s_3 The relations $\triangleleft$ and $\triangleright$ are mutually inverse: $a \triangleleft b \Leftrightarrow b \triangleright a$.
- s_4 The relations $\triangleleft$ and $\triangleright$ are transitive: $(a \triangleleft b) \wedge (b \triangleleft c) \Rightarrow a \triangleleft c$ and $(a \triangleright b) \wedge (b \triangleright c) \Rightarrow a \triangleright c$.
- s_5 The relations $\downarrow$ and $\uparrow$ are symmetric: $a \downarrow b \Leftrightarrow b \downarrow a$ and $a \uparrow b \Leftrightarrow b \uparrow a$.
- s_6 Triangular property: for $\sigma_1, \sigma_2, \sigma_3 \in \{\triangleleft \cup \triangleright, \downarrow, \uparrow\}$, it holds
- $$(a \sigma_1 b) \wedge (b \sigma_2 c) \wedge (c \sigma_3 a) \Rightarrow (\sigma_1 = \sigma_2) \vee (\sigma_2 = \sigma_3) \vee (\sigma_3 = \sigma_1) \quad .$$
- s_7 Square property:
- $s_7^{\triangleleft}$ $(a \triangleleft b) \wedge (a \triangleleft d) \wedge (c \triangleleft d) \Rightarrow (a \triangleleft c) \vee (b \triangleleft c) \vee (b \triangleleft d) \vee (c \triangleleft a) \vee (c \triangleleft b) \vee (d \triangleleft b) \quad ,$
- $s_7^{\downarrow}$ $(a \downarrow b) \wedge (a \downarrow d) \wedge (c \downarrow d) \Rightarrow (a \downarrow c) \vee (b \downarrow c) \vee (b \downarrow d) \quad ,$
- $s_7^{\uparrow}$ $(a \uparrow b) \wedge (a \uparrow d) \wedge (c \uparrow d) \Rightarrow (a \uparrow c) \vee (b \uparrow c) \vee (b \uparrow d) \quad .$

Proof The properties s_1, s_2, s_3, s_4 and s_5 are readily proved using the relevant definitions. Let us consider the more challenging cases of s_6 and s_7 .

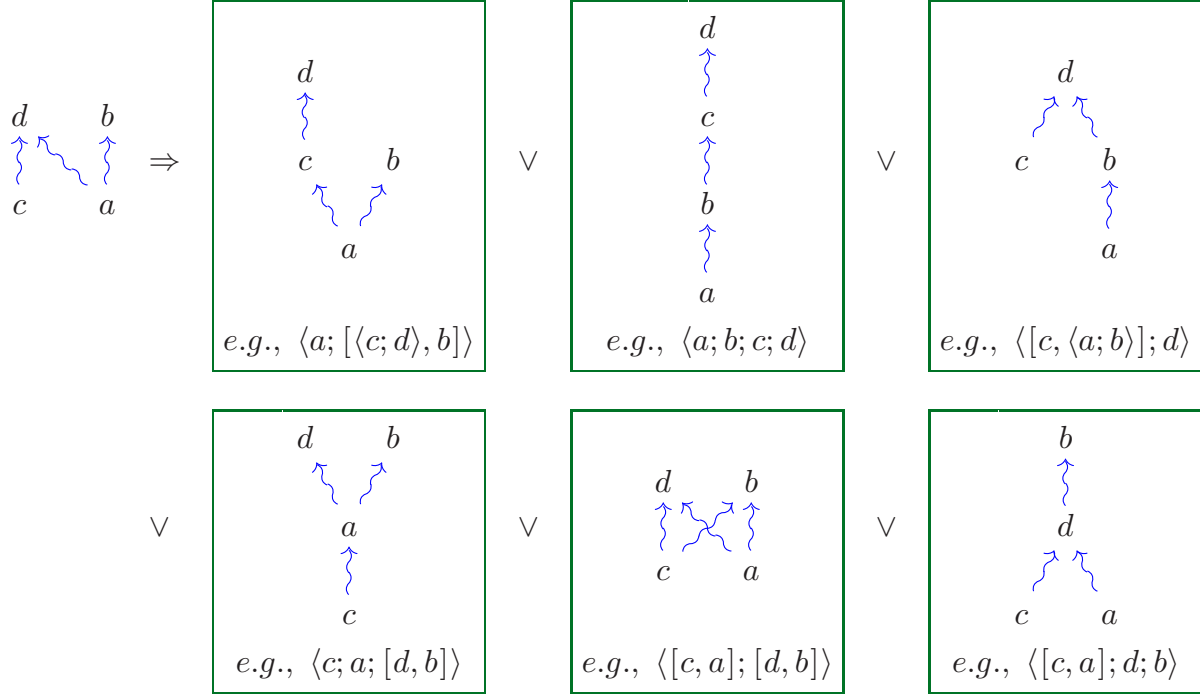
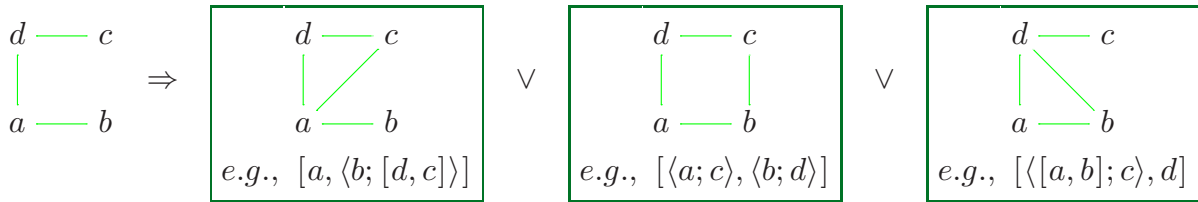
- s_6 Suppose that $a \triangleleft b$ and $b \downarrow c$: the only possible cases are $S\langle P\{a\}; T[Q\{b\}, R\{c\}] \rangle$ (so $a \triangleleft c$) and $S[T\langle P\{a\}; Q\{b\} \rangle, R\{c\}]$ (so $a \downarrow c$), for some structure contexts $P\{ \}$, $Q\{ \}$, $R\{ \}$, $S\{ \}$ and $T\{ \}$. Other combinations of σ_1 and σ_2 generate similar possibilities.
- s_7 Let us proceed by structural induction on S . Every structure with less than four atom occurrences trivially satisfies s_7 , since four distinct occurrences are requested, therefore let us consider situations where at least four atom occurrences are present. Let U and V be any two structures such that $U \neq \circ \neq V$ and either $S = \langle U; V \rangle$ or $S = [U, V]$ or $S = (U, V)$; let us choose a, b, c and d in S . If a, b, c and d are either all in U or all in V then we can apply the induction hypothesis; let us then consider the cases when they are not all in U or all in V . Consider $s_7^{\triangleleft}$ (Figure 2 should help). Since $S = [U, V]$ and $S = (U, V)$ falsify the hypothesis of $s_7^{\triangleleft}$, the only situation we have to consider is $S = \langle U; V \rangle$. Suppose that the conclusion of $s_7^{\triangleleft}$ is false and suppose that a is in U ; then c must be in U (otherwise $a \triangleleft c$ would be true), and then b and then d must be in U , but this contradicts our assumption. Analogously, if a is in V then c must be in V and then b and then d must be in V . In the end, if the hypothesis of $s_7^{\triangleleft}$ is true when a, b, c and d are scattered between U and V , then its conclusion is true. The same argument, simplified by the holding of symmetry, applies to $s_7^{\downarrow}$ and $s_7^{\uparrow}$. $\square$

The triangular property says that there are no structures such that the following configuration may be found in them:



in every triangle at least two sides must represent the same structural relation.

The square property for $\triangleleft$ may be represented like in Figure 2, where transitivity has been taken into account and an example structure is shown under each diagram. Enjoying commutativity, the cases of par and copar are simpler. See, in Figure 3, what happens with par. We can say, informally, that no square has exactly three sides or diagonals of the same nature and forming a simple path (disregarding orientation).

Fig. 2 Square property for $\triangleleft$ Fig. 3 Square property for $\downarrow$

2.2.5 Remark The relations $\downarrow$ and $\uparrow$ are not transitive: consider $[\langle a; c \rangle, b]$ and $(\langle a; c \rangle, b)$.

We saw that the conditions s_1 – s_7 are necessary for a structure. I am going to show that they are sufficient. The proof consists in recursively finding suitable partitions of a given set of atom occurrences ξ , where ξ is equipped with structural relations. If we are able to partition ξ into two parts μ and ν such that one and only one among $\mu \triangleleft \nu$, $\mu \downarrow \nu$ and $\mu \uparrow \nu$ holds, and we can go ahead like this recursively, then we can recover a structure. I need to establish some straightforward preliminary notions.

2.2.6 Definition A *web candidate* is a quadruple $\zeta = (\xi, \triangleleft, \downarrow, \uparrow)$, where ξ is a set of atom occurrences and $\triangleleft, \downarrow, \uparrow \subseteq \xi^2$. Given the web candidates $\zeta_\mu = (\mu, \triangleleft_\mu, \downarrow_\mu, \uparrow_\mu)$ and $\zeta_\nu = (\nu, \triangleleft_\nu, \downarrow_\nu, \uparrow_\nu)$, such that $\mu \neq \emptyset \neq \nu$, $\mu \cup \nu = \xi$ and $\mu \cap \nu = \triangleleft_\mu \cap \triangleleft_\nu = \downarrow_\mu \cap \downarrow_\nu = \uparrow_\mu \cap \uparrow_\nu = \emptyset$, the couple (ζ_μ, ζ_ν) can be:

1 a $\triangleleft$ -partition of ζ iff $\downarrow = \downarrow_\mu \cup \downarrow_\nu$, $\uparrow = \uparrow_\mu \cup \uparrow_\nu$ and

$$\triangleleft = \triangleleft_\mu \cup \triangleleft_\nu \cup \{ (a, b) \mid a \in \mu, b \in \nu \} \quad ;$$

2 a $\downarrow$ -partition of ζ iff $\triangleleft = \triangleleft_\mu \cup \triangleleft_\nu$, $\uparrow = \uparrow_\mu \cup \uparrow_\nu$ and

$$\downarrow = \downarrow_\mu \cup \downarrow_\nu \cup \{ (a, b) \mid (a \in \mu \wedge b \in \nu) \vee (a \in \nu \wedge b \in \mu) \} \quad ;$$

3 an $\uparrow$ -partition of ζ iff $\triangleleft = \triangleleft_\mu \cup \triangleleft_\nu$, $\downarrow = \downarrow_\mu \cup \downarrow_\nu$ and

$$\uparrow = \uparrow_\mu \cup \uparrow_\nu \cup \{ (a, b) \mid (a \in \mu \wedge b \in \nu) \vee (a \in \nu \wedge b \in \mu) \} \quad .$$

For every web candidate, the relation $\triangleright = \{ (a, b) \mid b \triangleleft a \}$ may be defined, and we will do so implicitly.

Of course, webs are (successful!) web candidates.

I am ready to show that s_1 – s_7 constitute an adequate characterisation of structures. It is important to note that these conditions can be checked over at most four atoms at a time, and this means that an algorithm that checks the conditions might be implemented locally in a network of processors representing a structure. I encourage the reader to check the following proof by trying to think at it as the task of a distributed algorithm.

2.2.7 Theorem *If the conditions s_1 – s_7 hold for a web candidate ζ then there is a structure whose web is ζ .*

Proof Let $\zeta = (\xi, \triangleleft, \downarrow, \uparrow)$: We will proceed by induction on the cardinality $|\xi|$ of ξ to build a structure S such that $\mathbf{w}S = \zeta$. If $\xi = \emptyset$ then $S = \circ$. If $\xi = \{a\}$ then $\triangleleft = \downarrow = \uparrow = \emptyset$ (by s_1) and $S = a$. Let us consider then the case where at least two atom occurrences are in ξ . We will see that the conditions s_1 – s_7 enforce the existence of a $\triangleleft$ -, $\downarrow$ - or $\uparrow$ -partition of ζ . Suppose that a $\triangleleft$ -partition of ζ exists, consisting of ζ_μ and ζ_ν . The conditions s_1 – s_7 hold for ζ_μ and ζ_ν , therefore, by induction hypothesis, two structures U and V exist such that $\mathbf{w}U = \zeta_\mu$ and $\mathbf{w}V = \zeta_\nu$. But then we can take $S = \langle U; V \rangle$, and, by definition and by s_2 and s_3 , we have $\mathbf{w}S = \zeta$. We can proceed analogously when ζ_μ and ζ_ν form a $\downarrow$ -partition (take $S = [U, V]$) or an $\uparrow$ -partition (take $S = (U, V)$); s_5 has a role here in ensuring the correct formation of a partition.

We have to show that a $\triangleleft$ -, $\downarrow$ - or $\uparrow$ -partition of ζ exists, in the given hypotheses, consisting of $\zeta_\mu = (\mu, \triangleleft_\mu, \downarrow_\mu, \uparrow_\mu)$ and $\zeta_\nu = (\nu, \triangleleft_\nu, \downarrow_\nu, \uparrow_\nu)$. We will construct the μ and ν sets of atom occurrences incrementally, starting from $\mu_2 = \{a\}$ and $\nu_2 = \{b\}$, for some a and b in ξ , and building a family of couples $\{(\mu_i, \nu_i)\}_{2 \leq i \leq |\xi|}$ such that at each step one element of ξ is added to the union of μ_i and ν_i that was not added before; at each step $\mu_i \neq \emptyset \neq \nu_i$ and either $\mu_i \triangleleft \nu_i$ or $\mu_i \downarrow \nu_i$ or $\mu_i \uparrow \nu_i$. The final step gives the partition, i.e., $\mu = \mu_{|\xi|}$ and $\nu = \nu_{|\xi|}$. Here is a non-deterministic algorithm.

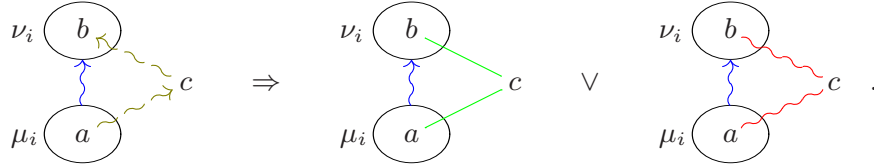
First Step

Take $\mu_2 = \{a\}$ and $\nu_2 = \{b\}$, where a and b are distinct atom occurrences that are randomly chosen in ξ and such that one of $\mu_2 \triangleleft \nu_2$, $\mu_2 \downarrow \nu_2$ or $\mu_2 \uparrow \nu_2$ holds (the conditions s_2 and s_3 apply).

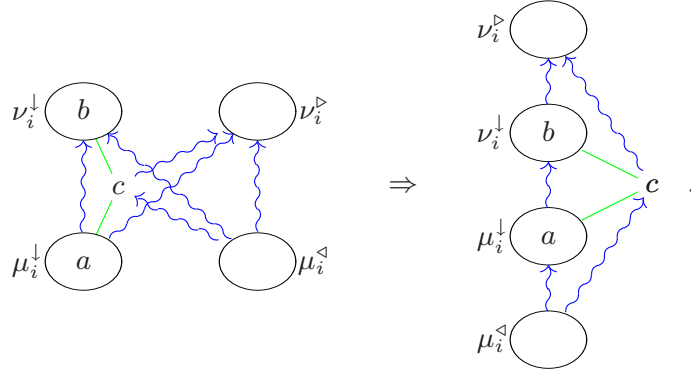
Iterative Step

We have two disjoint and non-empty sets of occurrences μ_i and ν_i such that all the atom occurrences in μ_i are in the same structural relation $\sigma \in \{\triangleleft, \downarrow, \uparrow\}$ with the atom occurrences in ν_i , i.e., either $\mu_i \triangleleft \nu_i$ or $\mu_i \downarrow \nu_i$ or $\mu_i \uparrow \nu_i$. Pick any c in $\xi \setminus (\mu_i \cup \nu_i)$. If $d \sigma c$ for every d in μ_i then let $\mu_{i+1} = \mu_i$ and $\nu_{i+1} = \nu_i \cup \{c\}$; if $c \sigma e$ for every e in ν_i then let $\mu_{i+1} = \mu_i \cup \{c\}$ and $\nu_{i+1} = \nu_i$; in both cases $\mu_{i+1} \sigma \nu_{i+1}$. Otherwise we have to rearrange μ_i and ν_i in order to meet our requirements. Let us proceed by cases.

- 1 $\mu_i \triangleleft \nu_i$ and there are a in μ_i and b in ν_i such that $\neg(a \triangleleft c)$ and $\neg(c \triangleleft b)$: this situation is represented at the left (where $a \rightsquigarrow c$ stands for $\neg(a \triangleleft c)$):

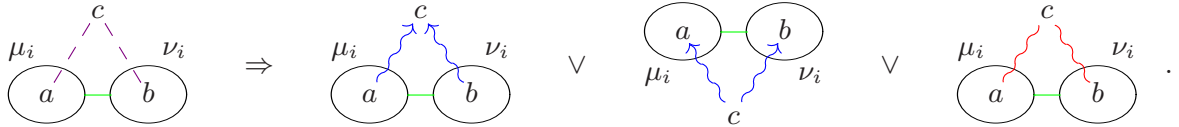


Since $a \triangleleft b$, by transitivity of $\triangleleft$ (s_4), by symmetry of $\downarrow$ and $\uparrow$ (s_5) and by the triangular property (s_6), only two cases are possible: either $a \downarrow c$ and $c \downarrow b$, or $a \uparrow c$ and $c \uparrow b$ (the former case is represented in the central diagram, the latter at the right). Let us only consider the first case, the other one being similar. Again by s_4 , s_5 and s_6 , either $d \downarrow c$ or $d \triangleleft c$, for each element d in μ_i , and either $c \downarrow e$ or $c \triangleleft e$, for each element e in ν_i . We can then partition μ_i into the two disjoint sets $\mu_i^\downarrow$ and $\mu_i^\triangleleft$ and ν_i into the two disjoint sets $\nu_i^\downarrow$ and $\nu_i^\triangleleft$ in such a way that $\mu_i^\downarrow \downarrow \{c\}$, $\mu_i^\triangleleft \triangleleft \{c\}$ and $\{c\} \downarrow \nu_i^\downarrow$, $\{c\} \triangleleft \nu_i^\triangleleft$; of course, $a \in \mu_i^\downarrow$ and $b \in \nu_i^\downarrow$. This situation is represented at the left:

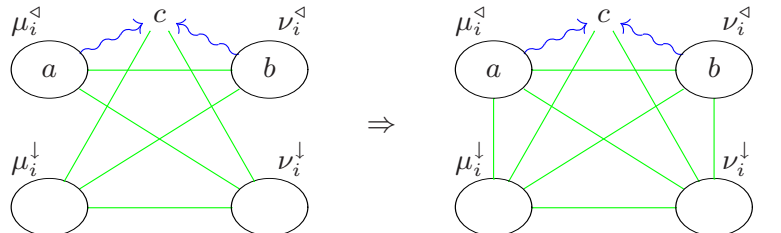


It must be the case that $\mu_i^\triangleleft \triangleleft \mu_i^\downarrow$, because of s_4 , s_5 , s_6 and the square property for $\downarrow$ ($s_7^\downarrow$) between $\mu_i^\downarrow$, $\mu_i^\triangleleft$, $\nu_i^\downarrow$ and c . Analogously, it must be $\nu_i^\downarrow \triangleleft \nu_i^\triangleleft$. The resulting situation, simplified by transitivity, is shown at the right. If $\mu_i^\triangleleft \neq \emptyset$ then take $\mu_{i+1} = \mu_i^\triangleleft$ and $\nu_{i+1} = \mu_i^\downarrow \cup \nu_i^\downarrow \cup \nu_i^\triangleleft \cup \{c\}$: in this case $\mu_{i+1} \triangleleft \nu_{i+1}$. Otherwise, if $\nu_i^\triangleleft \neq \emptyset$ then take $\mu_{i+1} = \mu_i^\downarrow \cup \nu_i^\downarrow \cup \{c\}$ and $\nu_{i+1} = \nu_i^\triangleleft$: again $\mu_{i+1} \triangleleft \nu_{i+1}$. If both $\mu_i^\triangleleft$ and $\nu_i^\triangleleft$ are empty, take $\mu_{i+1} = \mu_i^\downarrow \cup \nu_i^\downarrow$ and $\nu_{i+1} = \{c\}$: in this case $\mu_{i+1} \downarrow \nu_{i+1}$.

- 2 $\mu_i \downarrow \nu_i$ and there are a in μ_i and b in ν_i such that $\neg(a \downarrow c)$ and $\neg(b \downarrow c)$: By an analogous argument to that in Case 1, we have that this situation, represented at the left in the following diagram (where $a - - c$ stands for $\neg(a \downarrow c)$), entails the possibilities at the right, and those only:



Let us consider the first case, where $a \triangleleft c$ and $b \triangleleft c$. By use of the triangular property, we can partition μ_i into $\mu_i^\downarrow$ and $\mu_i^\triangleleft$ and ν_i into $\nu_i^\downarrow$ and $\nu_i^\triangleleft$ in such a way that we have the situation represented at the left:



The square property enforces the situation at the right, where we can now define an appropriate partition. If $\mu_i^\downarrow \neq \emptyset$ then take $\mu_{i+1} = \mu_i^\downarrow$ and $\nu_{i+1} = \mu_i^\triangleleft \cup \{c\} \cup \nu_i^\triangleleft \cup \nu_i^\downarrow$: in this case $\mu_{i+1} \downarrow \nu_{i+1}$. Otherwise, if $\nu_i^\downarrow \neq \emptyset$ then take $\mu_{i+1} = \mu_i^\triangleleft \cup \{c\} \cup \nu_i^\triangleleft$ and $\nu_{i+1} = \nu_i^\downarrow$: again $\mu_{i+1} \downarrow \nu_{i+1}$. If both $\mu_i^\downarrow$ and $\nu_i^\downarrow$ are empty, take $\mu_{i+1} = \mu_i^\triangleleft \cup \nu_i^\triangleleft$ and $\nu_{i+1} = \{c\}$: in this case $\mu_{i+1} \triangleleft \nu_{i+1}$. The other cases above, where $a \triangleright c$ and $b \triangleright c$ and where $a \uparrow c$ and $b \uparrow c$, are managed in an analogous way.

- 3** $\mu_i \uparrow \nu_i$ and there are a in μ_i and b in ν_i such that $\neg(a \uparrow c)$ and $\neg(b \uparrow c)$: this case is similar to Case 2.

The final step of the algorithm occurs when no atom occurrences are left in ξ that are not already in the partition. $\square$

In the previous proof, the reader might be puzzled by not seeing a ‘bottom-up’ construction where atoms are gathered together into bigger and bigger structures, starting from those that correspond to the smallest substructures of the structure to be built. The problem with that approach is that it is not symmetric: contrary to the approach above, only a few atoms can participate at any given time to the building of the structure. I feel that the symmetry in our construction might be important in expected future developments, aside from being a mathematically respectable property.

The core of the proof above relies on the combined action of the triangular and square properties. The property s_6 reduces the problem to a case in which only two structural relations are involved, then s_7 is used to decide the remaining undecided side of a square. This procedure can then be generalised to the case in which more than three different kinds of context are allowed in structures. In fact, going back to the proof of Theorem 2.2.4, it is easy to see that s_6 and s_7 do not depend on the number and quality of different structural relations; rather they make use of the uniqueness property s_2 , which is in turn guaranteed by the inherently unambiguous constitution of structures.

The algorithm given in the proof above is non-deterministic, therefore it cannot reasonably be used to answer the following, inevitable question: is a structure corresponding to a web unique? It turns out that it is, modulo equivalence of course, as the following theorem shows.

2.2.8 Lemma *Given a structure T , if (ζ_μ, ζ_ν) is a $\triangleleft$ -partition of $\mathsf{w}T$ (respectively, a $\downarrow$ -partition, an $\uparrow$ -partition) then there are two structures U and V such that $\mathsf{w}U = \zeta_\mu$, $\mathsf{w}V = \zeta_\nu$ and $T = \langle U; V \rangle$ (respectively, $T = [U, V]$, $T = (U, V)$).*

Proof Let $\mathsf{w}T = (\text{occ } T, \triangleleft, \downarrow, \uparrow)$; the web candidates $\zeta_\mu = (\mu, \triangleleft_\mu, \downarrow_\mu, \uparrow_\mu)$ and $\zeta_\nu = (\nu, \triangleleft_\nu, \downarrow_\nu, \uparrow_\nu)$ form a $\triangleleft$ -partition of $\mathsf{w}T$. Since $\mu \neq \emptyset \neq \nu$, the structure T falls in one of the three cases:

- 1** $T = \langle T_1; \dots; T_h \rangle$, where $h > 1$ and, for $1 \leq i \leq h$, it holds $T_i \neq \circ$ and T_i is not a proper seq: It must be the case that $\text{occ } T_i \subseteq \mu$ or $\text{occ } T_i \subseteq \nu$, for every i . In fact, suppose the contrary, and suppose that $T_i = [T'_i, T''_i]$ for some T'_i and T''_i such that $T'_i \neq \circ \neq T''_i$ (the same argument holds when $T_i = (T'_i, T''_i)$, in the same conditions). It is then possible to find a in T'_i and b in T''_i , or a in T''_i and b in T'_i , such that a is in μ and b is in ν . But then $a \downarrow b$, and this violates the hypothesis. Then, for every i , the atom occurrences in T_i come either from μ or from ν , but not from both. It must also be the case that there are k and $k+1$ in $1, \dots, h$ such that $T_1, \dots, T_k$ have all their atom occurrences in μ and $T_{k+1}, \dots, T_h$ have all their atom occurrences in ν , otherwise there would be cases of $b \triangleleft a$ for some a in μ and b in ν . Then take $U = \langle T_1; \dots; T_k \rangle$ and $V = \langle T_{k+1}; \dots; T_h \rangle$.
- 2** $T = [T', T'']$, where $T' \neq \circ \neq T''$: There must be a in T' and b in T'' , or a in T'' and b

in T' , such that a is in μ and b is in ν . But then $a \downarrow b$, and this violates the hypothesis. Therefore, this case is actually impossible.

3 $T = (T', T'')$, where $T' \neq \circ \neq T''$: The argument is the same as for Case 2.

Therefore, the lemma is proved for any $\lhd$ -partition of $\mathsf{w}T$. A similar argument holds for $\downarrow$ - and $\uparrow$ -partitions, made simpler by the fact that we should not worry about seq orders, as we had to do in Case 1 above. $\square$

2.2.9 Theorem *Two structures are equivalent if and only if they have the same web.*

Proof The ‘only if’ part is trivial, so let us concentrate on the ‘if’ one. Let S and T be two structures in normal form; we have to prove that if $\mathsf{w}S = \mathsf{w}T$ then $S = T$. Let us proceed by structural induction on S . In the base cases when $S = \circ$ or $S = a$ we trivially get $S = T$. Suppose now that there are P and Q such that $S = \langle P; Q \rangle$ and $P \neq \circ \neq Q$. The couple $(\mathsf{w}P, \mathsf{w}Q)$ is then a $\lhd$ -partition of $\mathsf{w}S$, and therefore of $\mathsf{w}T$. By Lemma 2.2.8 there are U and V such that $T = \langle U; V \rangle$ and $\mathsf{w}U = \mathsf{w}P$ and $\mathsf{w}V = \mathsf{w}Q$, and then the induction hypothesis applies. Similar arguments hold when $S = [P, Q]$ and $S = (P, Q)$, where $P \neq \circ \neq Q$. $\square$

3 Synthesis of a Formal System

In this section we will synthesise the formal system SBV, which is, as we will see, approximately a symmetric closure of system BV. Before plunging into the technicalities, and at the cost of some repetition, I believe it is useful to provide an informal account of what we will see later. Many of the intuitions valid in general for the calculus of structures can be seen here at work in the special case of the definition of system SBV.

We have by now a convenient syntax and an intuitive, albeit vague, ‘space-time’ interpretation. Getting back to our previous example, let us consider the problem of designing a system, equivalent to, or conservatively extending, multiplicative linear logic. We would also like this system to deal with structures without having the problem of the non-deterministic partitioning of the context seen above for $\otimes$. In the partitioning into two branches, double-premiss inference rules are doomed to lose some of the possibilities left open in the conclusion, which we want to retain. Let us see then if we can do this job with single-premiss rules. Dealing with copar under this constraint requires a big departure from Gentzen’s sequents style: In the sequent calculus a formula is decomposed in its main connective, and its context is split. In the calculus of structures a formula is moved into another formula in a way determined by their local structure, and their context stays fixed. The underlying claim, for which this paper provides some evidence, is that Gentzen’s sequent systems, and the tree structure of derivations in them, are perfectly fit for traditional logics, like LK for classical logic, but not necessarily for new, exotic ones, like the multiplicative core of linear logic.

I will better address this issue in the concluding remarks. For now let us just notice that the calculus of structures is rather radical regarding logical rules, i.e., rules applying to connectives: they completely disappear in favour of structural rules. Structures are expressive enough to internalise the tree organisation of a sequent derivation. Inference rules become more capable of controlling what happens in premisses, with respect to what rules can do in the sequent calculus, and having more control in inference rules yields a more efficient management of resources.

3.1 Definition An *inference rule* is a scheme of the kind

$$\rho \frac{T}{R} ,$$

where ρ is the *name* of the rule, T is its *premiss* and R is its *conclusion*; rule names are denoted by ρ and π . In an inference rule, either the premiss or the conclusion can be missing, but not both. When premiss and conclusion in an instance of an inference rule are equivalent, that instance is said *trivial*, otherwise it is said *non-trivial*. A (*formal*) *system* is a set of inference rules; formal systems are denoted by $\mathcal{S}$. A *derivation* in a certain formal system is a finite chain of instances of inference rules in the system, and it can consist of just one structure; derivations are denoted by Δ . The premiss of the topmost inference rule instance in a derivation, if present, is called the *premiss* of the derivation; if present, the conclusion of the bottommost inference rule instance is called the *conclusion* of the derivation; the premiss and conclusion of derivations consisting of one structure is that structure. A derivation Δ whose premiss is T , conclusion is R , and whose inference rules

are in $\mathcal{S}$ is indicated by $\Delta \parallel_{\mathcal{S}}^{\frac{T}{R}}$ (the name Δ can be omitted). The *length* of a derivation is the number of instances of inference rules appearing in it. Two systems $\mathcal{S}$ and $\mathcal{S}'$ are *strongly equivalent* if for every derivation $\parallel_{\mathcal{S}}^{\frac{T}{R}}$ there exists a derivation $\parallel_{\mathcal{S}'}^{\frac{T}{R}}$, and vice versa.

There are, in the sequent calculus, two complementary, dynamic views of derivations, which we can adapt to our case:

- 1 The *top-down* view: premisses join (in trees) to form new conclusions, and the derivation grows toward its conclusion; this can be called the *deductive* viewpoint.
- 2 The *bottom-up* view: the conclusion is the starting point and inference rules are used to reach the desired premisses; this can be called the *proof-construction* viewpoint.

For the moment, we should try not to assign any traditional, special meaning to the action of making a derivation grow upward or downward. Symmetry will be broken at last by the introduction of a logical axiom and, with it, of the concept of *proof*, which in this paper stays firmly in the tradition.

After choosing single-premiss rules, we are in a situation of extensive symmetry: derivations are chains of inferences (top-down symmetry), par and copar are the same kind of structure with two different names, seq is self-dual, and we have a symmetric intuitive interpretation for communication, meant as the possibility of interaction. This state of affairs is represented in Figure 4, where arrows stand for the direction of growth of a derivation (please take this picture only as an informal, intuitive suggestion). Interaction itself will of course involve negation (in linear logic it is realised in the identity logical axiom, for example). For the time being we deal with the structural relations seq, par and copar, which do not involve negation.

I will ‘distil’ logical rules out of conservation laws. Consider the following template

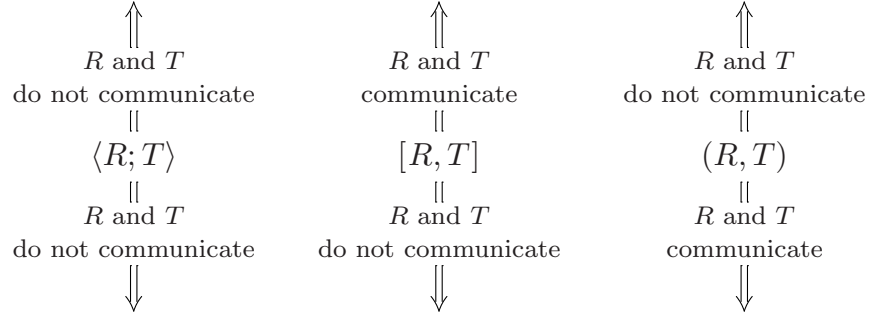


Fig. 4 *Communication along the direction of growth of a derivation*

of a rule, which I call *merge*:

$$\mathbf{g}\downarrow \frac{S\{Q\}}{S[R, T]} \quad ;$$

here we want to determine the laws which Q must obey. The rule says that if the structures R and T communicate in the context $S\{ \}$, then they can be replaced with the structure Q . How can we choose Q ? Why not say that Q is any structure made with all and only the atoms in R and T , and such that Q conserves all the structural relations that atoms have in R and T , and otherwise is free? Here two things are conserved:

- 1 the atoms, their number and polarity, and
- 2 their space-temporal relations in the two communicating structures.

It turns out that this almost works, but if we want cut elimination we have to add a third law of conservation, which preserves a sort of integrity. The law says that

- 3 if $a \uparrow b$ in R and $c \uparrow d$ in T , it cannot be in Q that $a \uparrow d$, $b \uparrow c$, $\neg(a \uparrow c)$ and $\neg(b \uparrow d)$.

Later on I will get back to this condition, which is probably obscure now. For the sake of symmetry, I also add its ‘cocondition’, in which $\uparrow$ is replaced with $\downarrow$.

The preceding laws find a natural definition by relation webs. We can consider a set, denoted by $R \diamond T$ and called *merge set*, where we collect all the structures Q that respect the conditions above. We know how to obtain these structures thanks to the characterisation by relation webs that we studied in the previous section. The rule $\mathbf{g}\downarrow$ is of course of limited practical use, because it requires picking up structures out of a large and difficult-to-compute set. Luckily, we can also characterise the merge set recursively, and then through a straightforward process of unfolding, we can obtain two simple rules that equivalently replace $\mathbf{g}\downarrow$: the *switch* ($\mathbf{s}$) and *seq* ($\mathbf{q}\downarrow$) rules.

There is another situation where communication occurs: between two structures R and T in a copar, while going downward in a derivation. I look then for explicit instances of this other rule, called *comerge*:

$$\mathbf{g}\uparrow \frac{S(R, T)}{S\{Q\}} \quad ,$$

where Q is again any structure in $R \diamond T$. After arguments symmetric to the ones used for the merge rule, I obtain two rules. One of them is again the switch rule, the other is new and is called *coseq* ($\mathbf{q}\uparrow$). Together the rules form system $\text{SBVc} = \{\mathbf{q}\downarrow, \mathbf{q}\uparrow, \mathbf{s}\}$. In SBVc , rule

$q\downarrow$ is the corule of $q\uparrow$ and s is its own corule. In a corule the premiss and the conclusion are negated and exchanged with respect to the corresponding rule. Still perfect top-down symmetry, as expected.

At this point we just have to add interaction rules, to form system SBV. They correspond to identity axioms and cut rules. There is no surprise in dealing with identity, but cut in the calculus of structures shows an interesting property. It is in fact possible, by using the switch and coseq rules, to replace its generic version by a version that only deals with atoms. This is a consequence of our insistence on maintaining a top-down symmetry. This fact entails two important properties:

- 1 it helps considerably in simplifying the cut elimination argument;
- 2 it separates the normal cut rule into two rules, one dealing with negation (so, interaction), and the other with order (or structure).

These properties are exhibited in all systems in the calculus of structures, and actually system SBV is the simplest example studied so far. The subsection on interaction insists on derivability results, which are responsible for the separation of the cut rule into ‘subrules’.

Here is the plan of the section: the merge rules will be derived from relation webs in the first subsection. In the second and third subsections we will derive what I call the ‘structure fragment’ of SBV. In the fourth subsection, we will study the ‘interaction fragment’ of SBV and some of its properties together with the structure one.

3.1 Merge Rules and Relation Webs

Intuitively, interacting means allowing the application of inference rules (with the final purpose of annihilating atoms). As we briefly mentioned already, the structures in a par context are allowed to interact along a derivation above the par, and cannot do so below the par. Dually, the structures in a copar context can interact going downward and cannot do so going upward. The structures in a seq context cannot interact either going up or going down in a derivation. The structures (R, T) and $[R, T]$ are then at the extremes of the possibilities of interaction between R and T . We can gather intermediate situations in the *merge set* of R and T .

In this subsection I will give two definitions of merge set. The first one is more ‘semantic’ in nature than the second one. I will prove that they are in fact equivalent, and the second definition will pave the way to the extraction of simple inference rules that compute the merge set, which is the subject of the next subsection. This subsection can be skipped by readers ignoring relation webs.

Let us firstly consider the following notion of *immersion* of a structure in another.

3.1.1 Definition We say that R is *immersed* in Q if:

- 1 the atom occurrences of R are atom occurrences of Q : $\text{occ } R \subseteq \text{occ } Q$;
- 2 the structural relations are respected: $\triangleleft_R \subseteq \triangleleft_Q$, $\downarrow_R \subseteq \downarrow_Q$, $\uparrow_R \subseteq \uparrow_Q$.

For example, all the structures immersed in $S = [\langle a; b \rangle, c]$ are: $\circ$, a , b , c , $\langle a; b \rangle$, $[a, c]$, $[b, c]$ and $[\langle a; b \rangle, c]$. The substructures of S are only $\circ$, a , b , c , $\langle a; b \rangle$ and $[\langle a; b \rangle, c]$.

3.1.2 Remark Given the structure Q , we can find all the structures immersed in it simply by choosing their atoms in Q in all the possible ways. Suppose the set $\xi \subseteq \text{occ } Q$ is chosen, and be $\mathbf{w} Q = (\text{occ } Q, \triangleleft, \downarrow, \uparrow)$. Consider the web candidate $\zeta = (\xi, \triangleleft \cap \xi^2, \downarrow \cap \xi^2, \uparrow \cap \xi^2)$: since $\mathbf{w} Q$ obeys s_1 – s_7 , so does ζ . By Theorem 2.2.7, a structure R exists such that $\mathbf{w} R = \zeta$; the structure R is immersed in Q , by definition, and is unique modulo equivalence by Theorem 2.2.9.

How can we compose two structures? The following definition is in part very natural, and in part motivated by our need of getting a cut elimination theorem in the end.

3.1.3 Definition Given two structures R and T such that $\text{occ } R \cap \text{occ } T = \emptyset$, the *merge set* $R \diamond T$ is the set of all the structures Q such that R and T are immersed in Q and:

$$\mathfrak{m}_1 \quad \text{occ } R \cup \text{occ } T = \text{occ } Q;$$

$\mathfrak{m}_2$ for all the distinct atom occurrences $a, b \in \text{occ } R$, and $c, d \in \text{occ } T$ the following hold in Q :

$$\mathfrak{m}_2^\downarrow \quad (a \downarrow b) \wedge (c \downarrow d) \wedge (a \downarrow d) \wedge (b \downarrow c) \Rightarrow (a \downarrow c) \vee (b \downarrow d) \quad ,$$

$$\mathfrak{m}_2^\uparrow \quad (a \uparrow b) \wedge (c \uparrow d) \wedge (a \uparrow d) \wedge (b \uparrow c) \Rightarrow (a \uparrow c) \vee (b \uparrow d) \quad .$$

When no confusion arises, we denote with $R \diamond T$ both the merge set and any of its elements.

The merge set $R \diamond T$ is obtained from R and T in a free way, provided that atoms, and the relations between them, are conserved. There are also further constraints imposed by the condition $\mathfrak{m}_2$. I do not have at this time an intuitive, *a priori* understanding of this condition, but I can offer a very important technical one, *a posteriori*: removing the condition $\mathfrak{m}_2$ leads to the impossibility of eliminating cuts.

It is maybe helpful visualising all the possibilities of interaction in the case, say, when $R = [a, b]$ and $T = [c, d]$. After ignoring symmetric cases due to permutations, four possibilities are allowed:

$$\begin{array}{ccccccc} \begin{array}{c} d \text{ --- } c \\ | \quad | \\ a \text{ --- } b \end{array} & , & \begin{array}{c} d \text{ --- } c \\ | \quad | \\ a \text{ --- } b \end{array} & , & \begin{array}{c} d \text{ --- } c \\ | \quad | \\ a \text{ --- } b \end{array} & \text{and} & \begin{array}{c} d \text{ --- } c \\ | \quad | \\ a \text{ --- } b \end{array} , \\ e.g., [a, b, c, d] & e.g., [a, \langle b; c \rangle, d] & e.g., [\langle [a, b]; c \rangle, d] & & e.g., \langle [a, b]; [c, d] \rangle \end{array}$$

where $b - - c$ means $\neg(b \downarrow c)$; below the diagrams, examples of structures in $R \diamond T$ are shown. The following case is ruled out by the square property $s_7^\downarrow$ (see Theorem 2.2.4; there is no structure having this representation):

$$\begin{array}{c} d \text{ --- } c \\ | \quad | \\ a \text{ --- } b \end{array} \quad .$$

The only case ruled out by the condition $\mathfrak{m}_2$ (specifically by $\mathfrak{m}_2^\downarrow$) is this:

$$\begin{array}{c} d \text{ --- } c \\ | \quad | \\ a \text{ --- } b \end{array} \quad \text{or, equivalently,} \quad \begin{array}{c} c \text{ --- } d \\ | \quad | \\ a \text{ --- } b \end{array} .$$

Therefore, for example, $[\langle a; c \rangle, \langle b; d \rangle] \notin R \diamond T$.

Both m_1 and m_2 can be considered conservation laws. The axiom m_1 enforces the strict application of the immersion relation: atoms are conserved together with the structural relations they had in the two structures that are merged.

To understand the axiom m_2 , consider for example $m_2^\uparrow$ as it applies to $R = (a, b)$ and $T = (c, d)$: it forbids the inclusion of $([a, c], [b, d])$ into $R \diamond T$. This means that, from the point of view of communication, R and T must be considered modules whose integrity can not be violated; either R enters into T or vice versa, but it cannot happen that a single module results from the communication of the two modules. This situation can be considered an *integrity conservation*.

3.1.4 Remark We could naturally extend the merge operator to sets of structures: Given the sets of structures ϕ and ψ , define $\phi \diamond \psi$ as the set $\{Q \mid \exists R \in \phi. \exists T \in \psi. (Q \in R \diamond T)\}$. Which are the properties of the law of composition $\diamond$? It turns out immediately that the set $\{\circ\}$ is a unit and $\diamond$ is commutative. Associativity does not hold: consider for instance $\{a\} \diamond (\{b\} \diamond \{(c, d)\}) \ni ([a, c], [b, d]) \notin (\{a\} \diamond \{b\}) \diamond \{(c, d)\}$.

Of course, Definition 3.1.3 is too implicit to be very helpful in a syntactic setting. I move one step more toward syntax with the following definition, which builds the merge set recursively.

3.1.5 Definition Given two structures R and T , the *merge set* $R \diamond T$ is recursively defined as the minimal set of structures such that:

- 1 $\langle R; T \rangle, \langle T; R \rangle, [R, T], (R, T) \in R \diamond T$;
- 2 for every R', R'', T', T'' such that $R = \langle R'; R'' \rangle$ and $T = \langle T'; T'' \rangle$, where $R' \neq \circ \neq R''$ or $T' \neq \circ \neq T''$, it holds

$$\{ \langle Q'; Q'' \rangle \mid Q' \in R' \diamond T', Q'' \in R'' \diamond T'' \} \subseteq R \diamond T \quad ;$$

- 3 for every R', R'' and T', T'' such that $R' \neq \circ \neq R''$ and $T' \neq \circ \neq T''$ the following hold:

- 1 if $R = [R', R'']$ then $\{ [Q', R''] \mid Q' \in R' \diamond T \} \subseteq R \diamond T$,
- 2 if $R = (R', R'')$ then $\{ (Q', R'') \mid Q' \in R' \diamond T \} \subseteq R \diamond T$,
- 3 if $T = [T', T'']$ then $\{ [T', Q''] \mid Q'' \in R \diamond T'' \} \subseteq R \diamond T$,
- 4 if $T = (T', T'')$ then $\{ (T', Q'') \mid Q'' \in R \diamond T'' \} \subseteq R \diamond T$.

I invite the reader to check the examples given above against this new definition of merge set. Also note the possible use of the unit in Case 3.1.5.2: Given $R = \langle R'; R'' \rangle$ and T , the merge set $R \diamond T$ contains, among others, the structures of the kind $\langle R' \diamond T; R'' \rangle$ and $\langle R'; R'' \diamond T \rangle$.

The effect of the condition m_2 on Definition 3.1.5 is to put a restriction on the way we build the merge set of a couple of structures whose external shape is that of a par or of a copar. Should the condition not be present, Definition 3.1.5 would be uniform for all

the three possible contexts. Instead, given for example $R = (R', R'')$ and $T = (T', T'')$, we cannot put $(R' \diamond T', R'' \diamond T'')$ in $R \diamond T$; we only are allowed to consider $(R' \diamond T, R'')$, $(R', R'' \diamond T)$, $(R \diamond T', T'')$ and $(T', R \diamond T'')$.

As promised, I am going to prove the equivalence of Definitions 3.1.3 and 3.1.5.

3.1.6 Theorem *Definitions 3.1.3 and 3.1.5 are equivalent.*

Proof Let us indicate with $\diamond_1$ and $\diamond_2$, respectively, the merge operators defined in 3.1.3 and 3.1.5. We have to prove that $Q \in R \diamond_1 T \Leftrightarrow Q \in R \diamond_2 T$, for any structures Q, R and T . Let us proceed by structural induction on Q .

Base Cases

- 1 $Q = \circ$: Then $Q \in R \diamond_1 T \Leftrightarrow (R = \circ) \wedge (T = \circ) \Leftrightarrow Q \in R \diamond_2 T$.
- 2 $Q = a$: Then $Q \in R \diamond_1 T \Leftrightarrow ((R = a) \wedge (T = \circ)) \vee ((R = \circ) \wedge (T = a)) \Leftrightarrow Q \in R \diamond_2 T$.

Inductive Cases

- 3 $Q = \langle Q'; Q'' \rangle$, where $Q' \neq \circ \neq Q''$:
 - $\Rightarrow$ There are structures R', R'' and T', T'' such that $R = \langle R'; R'' \rangle$, $T = \langle T'; T'' \rangle$, $\text{occ } R' \cup \text{occ } T' = \text{occ } Q'$ and $\text{occ } R'' \cup \text{occ } T'' = \text{occ } Q''$ (see Remark 3.1.2). Then $Q' \in R' \diamond_1 T'$ and $Q'' \in R'' \diamond_1 T''$. By induction hypothesis we have then $Q' \in R' \diamond_2 T'$ and $Q'' \in R'' \diamond_2 T''$, and then, by definition, $Q \in R \diamond_2 T$.
 - $\Leftarrow$ If $Q' = R$ and $Q'' = T$, or $Q' = T$ and $Q'' = R$, then Definition 3.1.3 holds, and then $Q \in R \diamond_1 T$. Otherwise there are R', R'' and T', T'' such that $R = \langle R'; R'' \rangle$, $T = \langle T'; T'' \rangle$ and $Q' \in R' \diamond_2 T'$, $Q'' \in R'' \diamond_2 T''$. By induction hypothesis, $Q' \in R' \diamond_1 T'$ and $Q'' \in R'' \diamond_1 T''$. Definition 3.1.3 then holds for Q , and we get $Q \in R \diamond_1 T$.
- 4 $Q = [Q', Q'']$, where $Q' \neq \circ \neq Q''$:
 - $\Rightarrow$ There are $Q_1, \dots, Q_h$ such that $h \geq 2$, $Q = [Q_1, \dots, Q_h]$ and, for $1 \leq i \leq h$, it holds $Q_i \neq \circ$ and Q_i is not a proper par. There can be no more than one Q_k between $Q_1, \dots, Q_h$ such that $\text{occ } Q_k \cap \text{occ } R \neq \emptyset \neq \text{occ } Q_k \cap \text{occ } T$. In fact, suppose that there are two distinct structures, say Q_k and Q_l , which enjoy this property: this leads to contradicting the condition $m_2^\downarrow$. We can suppose that $Q_k = (Q'_k, Q''_k)$ and $Q_l = (Q'_l, Q''_l)$ (it would be the same assuming $Q_k = \langle Q'_k; Q''_k \rangle$ or $Q_l = \langle Q'_l; Q''_l \rangle$, independently) and we can take a in Q'_k , b in Q'_l , c in Q''_k and d in Q''_l such that a and b are in R and c and d are in T . We would have then $a \downarrow b$, $c \downarrow d$, $a \downarrow d$, $b \downarrow c$, $a \uparrow c$ and $b \uparrow d$, and this is impossible. Therefore, there is no more than one Q_k between $Q_1, \dots, Q_h$ whose atom occurrences are partly in R and partly in T , and then, since $h \geq 2$, there is at least one Q_m whose atom occurrences are all either in R or in T . Let us suppose that they are in R (same argument when they are in T), i.e., $\text{occ } Q_m \subseteq \text{occ } R$. Let $\hat{Q} = [Q_1, \dots, Q_{m-1}, Q_{m+1}, \dots, Q_h]$, so that $Q = [\hat{Q}, Q_m]$; there is then R' such that $R = [R', Q_m]$ and $\text{occ } R' \subseteq \text{occ } \hat{Q}$, and by 3.1.3 it holds $\hat{Q} \in R' \diamond_1 T$. By induction hypothesis we have $\hat{Q} \in R' \diamond_2 T$ and then $Q \in R \diamond_2 T$ by Definition 3.1.5.
 - $\Leftarrow$ If $Q' = R$, $Q'' = T$ or $Q' = T$, $Q'' = R$ then Definition 3.1.3 holds trivially. Suppose instead that $R = [R', Q'']$ and $Q' \in R' \diamond_2 T$. By induction hypothesis $Q' \in R' \diamond_1 T$, and then $Q = [Q', Q''] \in R \diamond_1 T$. Consider in fact a and b in R and c and d in T , and let us check the condition $m_2^\downarrow$. If a and b are both in R' the condition holds because $Q' \in R' \diamond_1 T$. If a and b are both in Q'' then $a \downarrow c$ and $b \downarrow d$ are true because c and d are in Q' . If a is in R' and b is in Q'' then $b \downarrow d$. If a is in Q'' and b is in R' then $a \downarrow c$. The condition $m_2^\uparrow$ is easily checked to be valid. The same argument applies to the case when $T = [Q', T'']$ and $Q'' \in R \diamond_2 T''$.

5 $Q = (Q', Q'')$, where $Q' \neq \circ \neq Q''$: Analogous to Case 4. □

The merge set behaves well with respect to negation:

3.1.7 Theorem $S \in R \diamond T$ iff $\bar{S} \in \bar{R} \diamond \bar{T}$.

Proof Negation exchanges par and copar, and they are used symmetrically in the definitions of merge set. Moreover, it does not matter to the definitions of merge whether atoms are negative or not. □

3.2 The Structure Fragment

Remember that we consider communication the possibility to interact. It is natural to say that two structures which communicate can be interleaved: an element in their merge set is chosen. Merging two structures R and T eliminates of course many possibilities of interaction; while the relations in R and T are conserved, each relation between an atom in R and an atom in T needs not be. Making a derivation grow means narrowing, in the direction of the growth, the freedom of interacting (i.e., the amount of communication) of the structures involved.

3.2.1 Definition The following inference rules $\mathbf{g}\downarrow$ and $\mathbf{g}\uparrow$ are called respectively *merge* and *comerge*:

$$\mathbf{g}\downarrow \frac{S\{R \diamond T\}}{S[R, T]} \quad \text{and} \quad \mathbf{g}\uparrow \frac{S(R, T)}{S\{R \diamond T\}} \quad .$$

The system $\{\mathbf{g}\downarrow, \mathbf{g}\uparrow\}$ is called *merge system* $\mathbf{V}$, or $\mathbf{MV}$.

Used together, the two inference rules give rise to this particular situation, which clearly shows the intermediate position of the structures in the merge set:

$$\begin{array}{c} \mathbf{g}\uparrow \frac{S(R, T)}{S\{R \diamond T\}} \\ \mathbf{g}\downarrow \frac{S\{R \diamond T\}}{S[R, T]} \end{array} \quad .$$

The rules $\mathbf{g}\downarrow$ and $\mathbf{g}\uparrow$ are not very friendly: they involve the merge set, which is hard to obtain, in general. Moreover, they hide simple relations between different kinds of context. They can be replaced with simpler rules, and we can obtain these rules through a process of unfolding, by Definition 3.1.5.

3.2.2 Definition An inference rule ρ is *derivable* for the formal system $\mathcal{S}$ if $\rho \notin \mathcal{S}$ and for every instance $\rho \frac{T}{R}$ there exists a derivation $\frac{T}{R} \parallel \mathcal{S}$.

In Figure 5 the unfolding of $\mathbf{g}\downarrow$ is shown. The rule 1 ($\mathbf{g}\downarrow$) instantiates as rules 2, 3 and 4, according to the base in the inductive definition of merge set. The inductive cases are considered in instances 5, 6 and 7. The rules 8 and 9 are instances of 5 and 7, respectively. Now, the rules 2 and 4 are respectively instances of 8 and 9, so they are derivable for all the other ones; the rule 3 is vacuous; the rule 6 is equivalent to 1 (each is an instance of

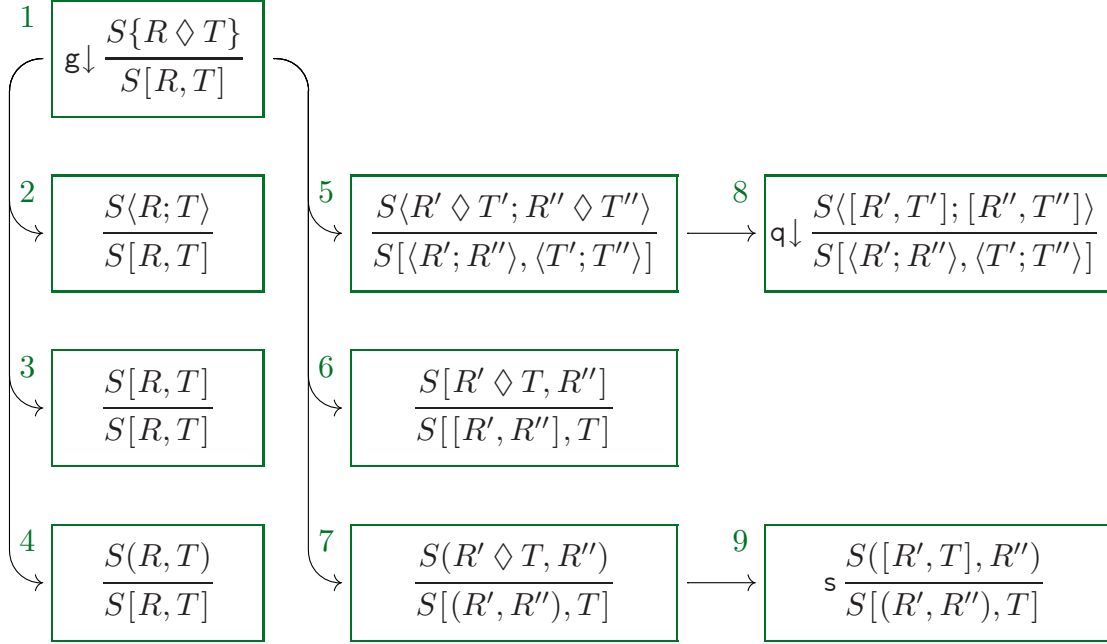


Fig. 5 Unfolding of the merge rule

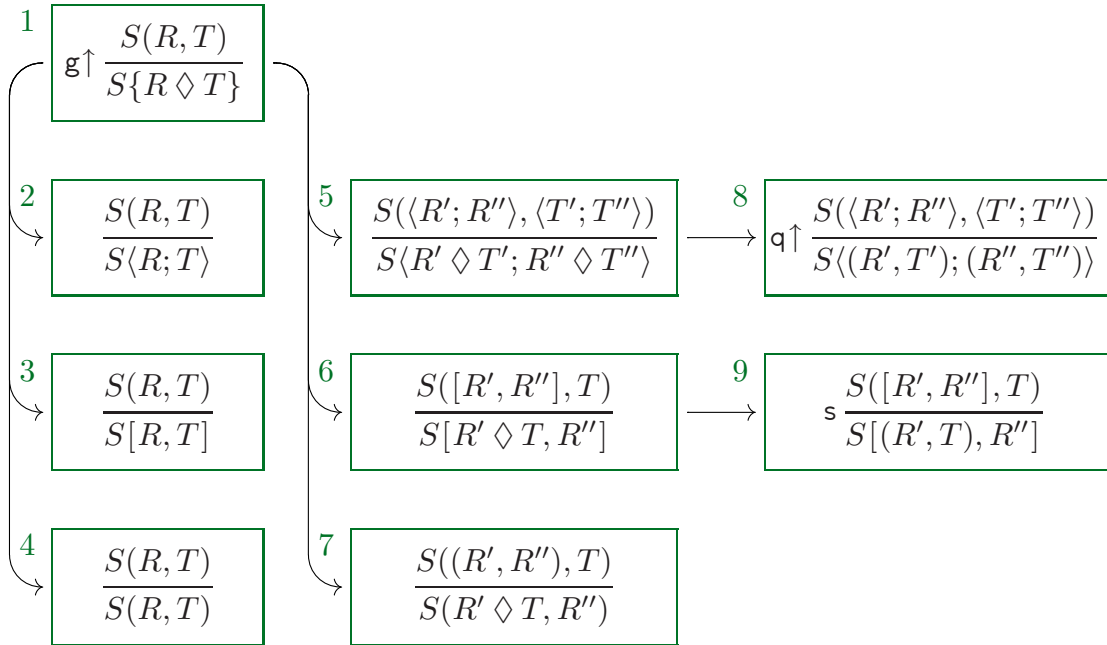


Fig. 6 Unfolding of the comerge rule

the other). In the end, $\text{g}\downarrow$ can be replaced with 5, 7, 8 and 9; but, since 5 is also derivable for $\{1, 8\}$, and 7 is derivable for $\{1, 9\}$, it must be the case that 1 is derivable for $\{8, 9\}$.

A dual argument holds for $\text{g}\uparrow$, and is shown in Figure 6. Please note that the rule 9 in Figure 6 is the same as the rule 9 in Figure 5.

The following theorems make the unfolding argument rigorous, but let us before give

a name to the newly discovered inference rules.

3.2.3 Definition The following inference rules $q\downarrow$, $q\uparrow$ and s are called respectively *seq*, *coseq* and *switch*:

$$q\downarrow \frac{S(\langle [R, T]; [R', T'] \rangle)}{S(\langle R; R' \rangle, \langle T; T' \rangle)} \quad , \quad q\uparrow \frac{S(\langle \langle R; T \rangle, \langle R'; T' \rangle \rangle)}{S(\langle (R, R') \rangle, \langle (T, T') \rangle)} \quad \text{and} \quad s \frac{S([R, T], R')}{S[(R, R'), T]} \quad .$$

The system $\{q\downarrow, q\uparrow, s\}$ is called *core symmetric basic system* $\vee$, or SBVc.

The core fragment in the system we are developing corresponds precisely to the structure fragment. In other systems, the core is a proper fragment of the structure one, in particular this is true for the extension of system BV studied in [10]. I refer the reader to that paper for a definition of core fragment, which I do not show here, since in our present, small system it would not make any discrimination.

3.2.4 Remark All the three rules in system SBVc can be used in the trivial way in which premiss and conclusion are equivalent, for example:

$$q\downarrow \frac{\langle [R, \circ]; [R', \circ] \rangle}{\langle \langle R; R' \rangle, \langle \circ; \circ \rangle \rangle} \quad , \quad q\uparrow \frac{\langle \langle \circ; T \rangle, \langle \circ; T' \rangle \rangle}{\langle \langle \circ, \circ \rangle, \langle T, T' \rangle \rangle} \quad \text{and} \quad s \frac{([R, T], \circ)}{[(R, \circ), T]} \quad .$$

Given three kinds of context (i.e., seq, par and copar), there are six possible ways in which a context is directly immersed into a different one. It turns out that each possible nesting appears as a premiss or as a conclusion in exactly one rule. The law for the creation of corules out of rules is clear at this point, and there is no surprise: premisses are exchanged with conclusions and submitted to negation. Please note that switch is its own corule.

The rule switch is common to the ‘down’ and ‘up’ fragments, and merge and comerge can be replaced with switch and the corresponding ‘down’ or ‘up’ seq rule.

3.2.5 Theorem *The rule $g\downarrow$ is derivable for $\{q\downarrow, s\}$.*

Proof Let $g\downarrow \frac{S\{Q\}}{S[R, T]}$ be an instance of $g\downarrow$, where $Q \in R \diamond T$. Let us proceed by structural induction on Q and use Definition 3.1.5.

Base Cases

- 1 $Q = \circ$: It must be $R = T = \circ$, then $S\{Q\} = S[R, T]$.
- 2 $Q = a$: It must be either $R = a$ and $T = \circ$, or $R = \circ$ and $T = a$; in either case $S\{Q\} = S[R, T]$.

Inductive Cases

- 3 $Q = \langle R; T \rangle$: Consider $q\downarrow \frac{S(\langle [R, \circ]; [\circ, T] \rangle)}{S(\langle R; \circ \rangle, \langle \circ; T \rangle)}$. The same argument holds when $Q = \langle T; R \rangle$.
- 4 $Q = [R, T]$: It holds $S\{Q\} = S[R, T]$.
- 5 $Q = (R, T)$: Consider $s \frac{S(\langle \circ, R \rangle, T)}{S[(\circ, T), R]}$.

Suppose now that no case between the preceding ones applies. Suppose that there are Q' and Q'' such that $Q' \neq \circ \neq Q''$:

- 6** $Q = \langle Q'; Q'' \rangle$: There must be R', R'', T', T'' such that $Q' \in R' \diamond T', Q'' \in R'' \diamond T''$ and $R = \langle R'; R'' \rangle, T = \langle T'; T'' \rangle$. Apply the induction hypothesis on

$$\begin{array}{c} \text{g}\downarrow \frac{S\langle Q'; Q'' \rangle}{S\langle Q'; [R'', T''] \rangle} \\ \text{g}\downarrow \frac{S\langle [R', T']; [R'', T''] \rangle}{S\langle [R', T']; \langle R'', T'' \rangle \rangle} \\ \text{q}\downarrow \frac{S\langle [R', T']; \langle R'', T'' \rangle \rangle}{S[\langle R'; R'' \rangle, \langle T'; T'' \rangle]} \end{array} .$$

- 7** $Q = [Q', Q'']$: There must be either R' such that $R = [R', Q'']$ and $Q' \in R' \diamond T$, or T'' such that $T = [Q', T'']$ and $Q'' \in R \diamond T''$. Apply the induction hypothesis on

$$\text{g}\downarrow \frac{S[Q', Q'']}{S[R', T, Q'']} \quad \text{or} \quad \text{g}\downarrow \frac{S[Q', Q'']}{S[Q', R, T'']} .$$

- 8** $Q = (Q', Q'')$: There must be either R' such that $R = (R', Q'')$ and $Q' \in R' \diamond T$, or T'' such that $T = (Q', T'')$ and $Q'' \in R \diamond T''$. Apply the induction hypothesis on

$$\begin{array}{c} \text{g}\downarrow \frac{S(Q', Q'')}{S([R', T], Q'')} \\ \text{s} \frac{S([R', T], Q'')}{S[(R', Q''), T]} \end{array} \quad \text{or} \quad \begin{array}{c} \text{g}\downarrow \frac{S(Q', Q'')}{S(Q', [R, T''])} \\ \text{s} \frac{S(Q', [R, T''])}{S[R, (Q', T'')]} \end{array} .$$

□

The ‘coproof’ of the last theorem is a proof of the following one.

3.2.6 Theorem *The rule $\text{g}\uparrow$ is derivable for $\{\text{q}\uparrow, \text{s}\}$.*

3.2.7 Theorem *The systems MV and SBVc are strongly equivalent.*

Proof It immediately follows from Theorems 3.2.5 and 3.2.6 and the fact that the rules of SBVc are instances of those of MV. □

Figure 7 shows the three rules in their graphical representation. To get an intuitive understanding of our rules, it is better to stick to a particular viewpoint: let us consider derivations from the bottom-up, proof-construction perspective.

The rule $\text{q}\downarrow$ models the mutual behaviour of seq and par, i.e., sequentialisation and communication: two seq structures that can interact can do so in their parts, provided that the order is respected. The rule also puts a limitation in the way the context can interact with two structures in a seq. Consider $\langle [R; R'], \langle T; T' \rangle \rangle$: structures R and R' do not communicate, but situations where R communicates with $\langle T; T' \rangle$, or R' with $\langle T; T' \rangle$, or R with T and R' with T' are possible:

$$\text{q}\downarrow \frac{S\langle [R, \langle T; T' \rangle]; [R', \circ] \rangle}{S[\langle R; R' \rangle, \langle \langle T; T' \rangle; \circ \rangle]} \quad , \quad \text{q}\downarrow \frac{S\langle [R, \circ]; [R', \langle T; T' \rangle] \rangle}{S[\langle R; R' \rangle, \langle \circ; \langle T; T' \rangle \rangle]} \quad , \quad \text{q}\downarrow \frac{S\langle [R, T]; [R', T'] \rangle}{S[\langle R; R' \rangle, \langle T; T' \rangle]} .$$

On the contrary, it is impossible for R to interact with T' and at the same time for R' to interact with T . A notable instance of $\text{q}\downarrow$ is

$$\text{q}\downarrow \frac{S\langle R; T \rangle}{S[R, T]} :$$

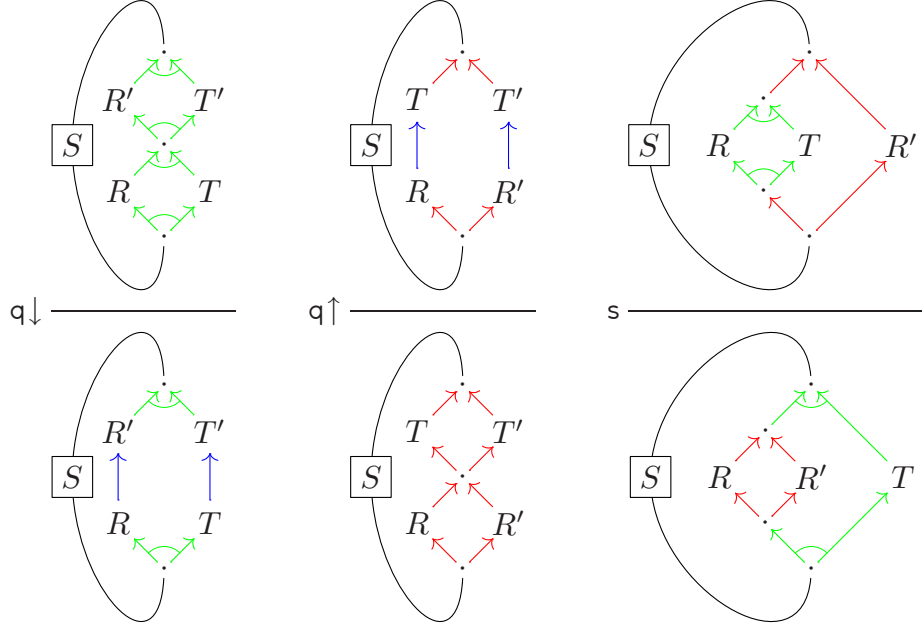


Fig. 7 Seq, coseq and switch rules (system SBVc)

if two non-communicating structures R and T can be proved in $S\{ \}$, *a fortiori* they can be proved in the same context when they communicate.

A good test for checking one's understanding of the $q\downarrow$ rule is verifying that the following is one of its instances:

$$q\downarrow \frac{[\langle [R; T], P', U' \rangle; P''], U'']}{[\langle R; T \rangle, \langle P'; P'' \rangle, U', U'']}. \quad .$$

We will meet again this and other similar difficult cases in the proof of the splitting theorem.

The rule s models the mutual behaviour of par and copar. A good way of understanding this rule is by considering the times rule of linear logic:

$$\otimes \frac{\vdash A, \Phi \quad \vdash B, \Psi}{\vdash A \otimes B, \Phi, \Psi},$$

where A and B are formulae and Φ and Ψ are multisets of formulae. In our system the par context plays the role of the comma in linear logic's right side of sequents (so, it is essentially the connective $\wp$), and copar plays the role of the connective $\otimes$. The two branches in the derivation generated by an application of the rule $\otimes$ do not communicate in a proof construction. By applying the rule, one has to choose which part of the context communicates with which formula in the times relation: in our case Φ can interact with A and Ψ with B . The difference between linear logic and our case is that in linear logic the rule $\otimes$ compels an early choice of the partition of the context, while our system does not. The same situation in the times rule above is realised like this, by using switch:

$$\begin{array}{c} s \frac{([R_A, T_\Phi], [R'_B, U_\Psi])}{[(R_A, T_\Phi), R'_B], U_\Psi]} \\ s \frac{[(R_A, T_\Phi), R'_B], U_\Psi]}{[(R_A, R'_B), T_\Phi, U_\Psi]} \end{array} .$$

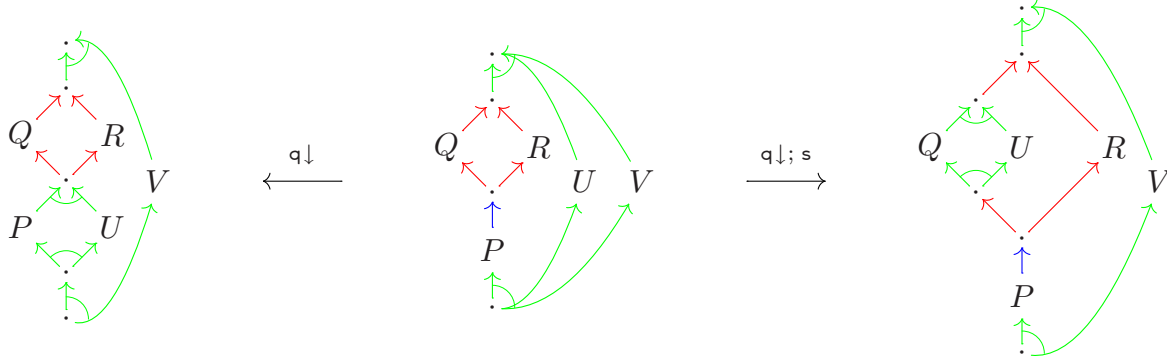


Fig. 8 Two possible rewritings of the central structure in a proof search

Of course the two applications of s are independent, so a certain part of the context can be brought to communicate inside a copar on demand during the search for a proof, which is something that cannot be done inside the sequent system of linear logic. A notable instance of s is

$$s \frac{S(R, T)}{S[R, T]} :$$

it is of course reminiscent of the rule mix of linear logic, especially when $S\{\circ\} = \circ$. I will formally investigate the relation between our system and linear logic in Section 5.

The use of units makes these rules particularly flexible: the example in Figure 8 illustrates the use of the rules $q\downarrow$ and s in proof construction, i.e., when considering the rewriting of structures going upward in a derivation. In the central structure of the figure, both U and V are applicable to the entire time span of the structure. Going upward in the derivation means *deciding, making choices* about which substructure interacts with which other. In the left structure, U has been brought in communication with P (after a single application of $q\downarrow$); in the right one, it has been brought in communication with Q (after an application of $q\downarrow$ and one of s). A seq maintains ordering, a copar acts as a switch for the structures coming from the outside.

At this point, a good test for the reader is trying to prove Proposition 4.2.2.

The rule $q\uparrow$ models the mutual behaviour of copar and seq. In $S\langle(R, R'); (T, T')\rangle$ no substructure between R, R', T and T' communicates with any other one. Rewriting that structure to $S\langle\langle R; T \rangle, \langle R'; T' \rangle\rangle$ does not make a difference in this respect, but limits the ability of the substructures in $S\{ \}$ to interact with R, R', T and T' . For example, given in $S\{ \}$ the structure $\langle R''; T'' \rangle$ that is in communication with $\langle(R, R'); (T, T')\rangle$, it would be possible to reach a state where R'' communicates with R and T'' communicates with T' . This is no longer possible after rewriting $\langle(R, R'); (T, T')\rangle$ to $\langle\langle R; T \rangle, \langle R'; T' \rangle\rangle$: in the latter structure R and T' fall into a copar relation. A notable instance of $q\uparrow$ is

$$q\uparrow \frac{S(R, T)}{S\langle R; T \rangle} .$$

After establishing that it is intuitively sound, we have to ask ourselves whether $q\uparrow$ adds something to $\{g\downarrow\}$ or is derivable.

It turns out immediately that $q\uparrow$ is not derivable for $\{q\downarrow, s\}$. Just consider the following derivation, which cannot be mimicked in $\{q\downarrow, s\}$:

$$q\uparrow \frac{(\langle a; c \rangle, b)}{\langle (a, b); c \rangle} .$$

Analogously, the derivation

$$s \frac{([a, c], b)}{[(a, b), c]}$$

can be used to show that s is not derivable for $\{q\downarrow, q\uparrow\}$.

We will achieve admissibility of $q\uparrow$, for a stronger system than $\{q\downarrow, s\}$, but we have first to treat negation, by introducing rules for interaction. Until now we have been dealing with structures without looking at the polarity of atoms. We were not interested in knowing if a certain atom was a or b or $\bar{a}$, but just in the fact that a certain place contained an atom. Interaction will consider the whole information carried by atoms, by realising the communication between atoms that are dual through negation. I will deal with it in Subsection 3.4.

3.3 Comments on the Structure Fragment

System SBVc is the combinatorial core of the system we are building: it is about maintaining *order* (in a broader sense than the usual mathematical notion). In [21] Retoré studies properties of proof nets for pomset logic, an extension of linear logic with a non-commutative connective, derived from coherence semantics. He shows a rewriting system that yields cut elimination for certain proof nets. That rewriting system is essentially equivalent to system SBVc. Retoré got his rewriting rules from a study of the inclusion relation for series-parallel orders, in [4]. Since series-parallel orders are at the basis of my definition of structures, Retoré's result and mine are similar because they come from the same underlying combinatorial phenomena. The difference between his work and mine is in the use we make of the rewriting system: he applies its rules in the cut elimination procedure, I use them directly to obtain inference rules.

It would be interesting to characterise derivability in SBVc and MV in terms of structural relations. More precisely: There is a natural ordering relation, due to Retoré, $<$ between $\lhd$, $\downarrow$ and $\uparrow$, namely $\uparrow < \lhd < \downarrow$. The relation $<$ compares the 'energy' of structures, i.e., it compares the number of interactions that they make possible. The more one goes up in a derivation, the more choices one makes, the fewer possibilities of interaction are left. From this point of view the goal of a search for a proof is 'cooling down' a structure to zero.

3.3.1 Definition Between structures, the partial order relation $\leq$ is defined such that $T \leq R$ iff $\text{occ } T = \text{occ } R$, $\downarrow_T \subseteq \downarrow_R$ and $\uparrow_T \supseteq \uparrow_R$.

I think that this definition is adequate to characterise derivability in a weaker version of MV, one that disallows cut elimination:

3.3.2 Definition Let WMV (*weak merge system* V) be the system $\{\mathbf{wg}\downarrow, \mathbf{wg}\uparrow\}$, where $\mathbf{wg}\downarrow$ and $\mathbf{wg}\uparrow$ are like $\mathbf{g}\downarrow$ and $\mathbf{g}\uparrow$ except that the definition of merge set adopted is 3.1.3 where $\mathbf{m}_2$ is not required.

System WMV is then like MV, except that the integrity conservation of par and copar is not enforced. I could show that WMV is strongly equivalent to $\{\mathbf{q}\downarrow, \mathbf{q}\uparrow, \mathbf{ws}\}$, where $\mathbf{ws}$

is $\mathbf{ws} \frac{S([R, T], [R', T'])}{S[(R, R'), (T, T')]} \cdot$ I believe that the following statement is true:

3.3.3 Conjecture $T \leq R$ iff there exists $\frac{T}{R} \parallel_{\text{WMV}}$.

At this time I do not see an easy argument to prove it, and this paper is already very long. Anyway, what is really interesting is:

3.3.4 Problem *Finding an improved definition of $\leq$, by adding conditions based only on webs, such that Conjecture 3.3.3 holds for MV, instead of WMV.*

There are chances that the necessary, new condition on webs would teach us something about cut elimination, since it would correspond to having the property $\mathbf{m}_2$, which has in turn been introduced for having cut elimination.

Definition 3.3.1 is already useful:

3.3.5 Remark In each non-trivial rule instance $\mathbf{q}\downarrow \frac{T}{R}$, $\mathbf{q}\uparrow \frac{T}{R}$ and $\mathbf{s} \frac{T}{R}$ we have $T < R$ (and $T = R$ holds if the rule instances are trivial, see Remark 3.2.4).

3.3.6 Theorem *For a given structure S , the set of derivations in SBVc, whose conclusion is S and where no trivial rule instance appears, is finite; the same holds for the similar set of derivations whose premiss is S .*

Proof It immediately follows from Remark 3.3.5, the fact that the set of structures whose atom occurrences are the same as those of S is finite, and the fact that only a finite number of instances of rules is applicable to each structure. (Remember that we consider structures as equivalence classes.) $\square$

3.4 Interaction

3.4.1 Definition The following two inference rules $\mathbf{i}\downarrow$ and $\mathbf{i}\uparrow$ are called respectively *interaction* and *cointeraction* (or *cut*):

$$\mathbf{i}\downarrow \frac{S\{\circ\}}{S[R, \bar{R}]} \quad \text{and} \quad \mathbf{i}\uparrow \frac{S(R, \bar{R})}{S\{\circ\}} \quad .$$

The following two inference rules $\mathbf{ai}\downarrow$ and $\mathbf{ai}\uparrow$ are called respectively *atomic interaction* and *atomic cointeraction* (or *atomic cut*):

$$\mathbf{ai}\downarrow \frac{S\{\circ\}}{S[a, \bar{a}]} \quad \text{and} \quad \mathbf{ai}\uparrow \frac{S(a, \bar{a})}{S\{\circ\}} \quad .$$

Interaction is a natural law of conservation. In atomic interaction (which of course corresponds to an identity axiom in the sequent calculus), two complementary atoms in

direct communication annihilate each other and disappear at the same time from the structure in which they are immersed. We can think of ‘charge’ as being conserved. We can add this rule to system SBVc with no hesitation, since we are now able to produce derivations for the structure $[a, b, (\bar{a}, [\bar{b}, c), \bar{c}]]$ that correspond to its proof in linear logic, and without the context partitioning problem. Consider for example the following two possibilities:

$$\begin{array}{c}
 \text{ai} \downarrow \frac{\text{o} \frac{}{[a, \bar{a}]}}{[a, (\bar{a}, [c, \bar{c}])]} \\
 \text{ai} \downarrow \frac{}{[a, (\bar{a}, c), \bar{c}]} \\
 \text{s} \frac{}{[a, ([b, \bar{b}], [(\bar{a}, c), \bar{c}])]} \\
 \text{ai} \downarrow \frac{}{[a, b, (\bar{b}, [(\bar{a}, c), \bar{c}])]} \\
 \text{s} \frac{}{[a, b, (\bar{b}, [(\bar{a}, c), \bar{c}])]}
 \end{array}
 \quad \text{and} \quad
 \begin{array}{c}
 \text{ai} \downarrow \frac{\text{o} \frac{}{[b, \bar{b}]}}{[b, (\bar{b}, [a, \bar{a}])]} \\
 \text{ai} \downarrow \frac{}{[a, b, (\bar{b}, \bar{a})]} \\
 \text{s} \frac{}{[a, b, (\bar{b}, \bar{a}, [c, \bar{c}])]} \\
 \text{ai} \downarrow \frac{}{[a, b, (\bar{b}, [(\bar{a}, c), \bar{c}])]} \\
 \text{s} \frac{}{[a, b, (\bar{b}, [(\bar{a}, c), \bar{c}])]}
 \end{array}
 .$$

In these two derivations the atoms to be annihilated are brought in direct communication on demand. Notice also that in the derivation at the right the first atoms that are annihilated are c and $\bar{c}$, and they are nested inside the conclusion structure.

As in all the well-behaved systems, the rule $\text{i} \downarrow$ can be replaced by $\text{ai} \downarrow$, $\text{q} \downarrow$ and s , as we will see easily. This way interaction between complex structures is reduced to the atomic one by use of the combinatorial machinery we already have.

The cut rule of linear logic, represented in our syntax below at the left, can be realised by the derivation at the right:

$$\frac{([P, R], [\bar{R}, Q])}{[P, Q]} \quad \text{is obtained by} \quad \frac{\frac{\frac{([P, R], [\bar{R}, Q])}{[Q, ([P, R], \bar{R})]} \text{s} \frac{([P, R], [\bar{R}, Q])}{[P, Q, (R, \bar{R})]} \text{s} \frac{([P, R], [\bar{R}, Q])}{[P, Q]}}{[P, Q]} \text{i} \uparrow .$$

Since we live in a symmetric world, it is not surprising that cut admits a symmetric factorisation with respect to the interaction rule: cut can equivalently be replaced with the system $\{\text{ai} \uparrow, \text{q} \uparrow, \text{s}\}$. In atomic cointeraction, at a certain point going forward in time, a particle and an anti-particle pop up out of empty space. Contrarily to what seems to be happening in nature, they cannot immediately annihilate each other in order for energy to be conserved. This process, which in our system would lead to not having cut elimination, is forbidden by the axiom m_2 on integrity (see Definition 3.1.3). In fact, consider

$$\begin{array}{c}
 \text{ai} \downarrow \frac{\text{o} \frac{}{[e_a, \bar{e}_c]}}{([e_a, \bar{e}_c], [\bar{e}_b, e_d])} \\
 \text{ai} \downarrow \frac{}{[(e_a, \bar{e}_b), (\bar{e}_c, e_d)]} \\
 \text{ws} \frac{}{[e_a, \bar{e}_b]} \\
 \text{ai} \uparrow \frac{}{[e_a, \bar{e}_b]} \\
 \text{ai} \uparrow \frac{\text{o} \frac{}{[e_a, \bar{e}_b]}}{[e_a, \bar{e}_b]} ,
 \end{array}$$

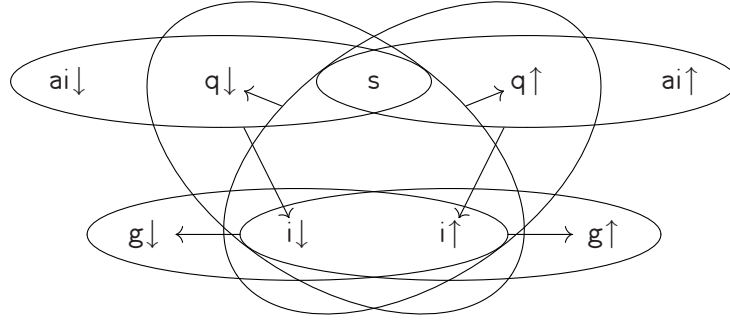


Fig. 9 Some derivability relations

where ws is a ‘weak’ switch instance that does not comply with the conservation law on integrity (the atoms have been indexed with a, b, c and d to help checking the condition). Here energy would be overall conserved, but there would also be a proof of $(e, \bar{e})$ that makes an essential use of cut: interaction between e and $\bar{e}$ is indirectly allowed by the use of two auxiliary atoms, which must be created *ad hoc*. Therefore, it is mandatory, for having cut elimination, that copar structures are introduced as a whole into the cut’s copar context, and this is precisely what the axiom m_2 guarantees. What is conserved is the integrity of all the copars in a communication: either one enters another, or vice versa. It is forbidden that any two of them are broken and rebuilt into a larger copar, which prevents from communicating pieces of both the original copars, which are in turn in mutual communication (or sequential composition).

Of course, $ai\downarrow$ and $ai\uparrow$ are instances of $i\downarrow$ and $i\uparrow$. The first interesting question about $i\downarrow$ and $i\uparrow$ is whether they are derivable, respectively, for $\{ai\downarrow, g\downarrow\}$ and $\{ai\uparrow, g\uparrow\}$, i.e., whether they are derivable for $\{ai\downarrow, q\downarrow, s\}$ and $\{ai\uparrow, q\uparrow, s\}$. Several relations of derivability will be given in the following, therefore I provide a map in Figure 9, where $(\mathcal{S}) \rightarrow \rho$ means that the rule ρ is derivable for the system $\mathcal{S}$.

3.4.2 Theorem *The rule $i\downarrow$ is derivable for $\{ai\downarrow, q\downarrow, s\}$.*

Proof Given $i\downarrow \frac{S\{\circ\}}{S[R, \bar{R}]}$, let us perform a structural induction on R .

Base Cases

- 1 $R = \circ$: In this case $S[R, \bar{R}] = S\{\circ\}$.
- 2 R is an atom: Then the given instance of $i\downarrow$ is an instance of $ai\downarrow$.

Inductive Cases

- 3 $R = \langle P; Q \rangle$, where $P \neq \circ \neq Q$: Apply the induction hypothesis on $q\downarrow \frac{i\downarrow \frac{S\{\circ\}}{S[Q, \bar{Q}]} \quad S\langle [P, \bar{P}]; [Q, \bar{Q}] \rangle}{S[\langle P; Q \rangle, \langle \bar{P}; \bar{Q} \rangle]}$.
- 4 $R = [P, Q]$, where $P \neq \circ \neq Q$: Apply the induction hypothesis on $s \frac{i\downarrow \frac{S\{\circ\}}{S[Q, \bar{Q}]} \quad S([P, \bar{P}], [Q, \bar{Q}])}{S[Q, ([P, \bar{P}], \bar{Q})]} \quad S[P, Q, (\bar{P}, \bar{Q})]}$.
- 5 $R = (P, Q)$, where $P \neq \circ \neq Q$: Analogous to Case 4. □

3.4.3 Theorem *The rule $i\uparrow$ is derivable for $\{ai\uparrow, q\uparrow, s\}$.*

Proof The argument is symmetric to the one in the proof of Theorem 3.4.2. Let us just see the two derivations involved in the inductive cases, where $P \neq \circ \neq Q$:

$$\begin{array}{c}
 q\uparrow \frac{S(\langle P; Q \rangle, \langle \bar{P}; \bar{Q} \rangle)}{i\uparrow \frac{S(\langle P, \bar{P} \rangle; \langle Q, \bar{Q} \rangle)}{i\uparrow \frac{S(Q, \bar{Q})}{S\{\circ\}}}} \\
 \text{and} \\
 s \frac{S(P, Q, [\bar{P}, \bar{Q}])}{s \frac{S(Q, [(P, \bar{P}), \bar{Q}])}{s \frac{S[(P, \bar{P}), (Q, \bar{Q})]}{i\uparrow \frac{S(Q, \bar{Q})}{S\{\circ\}}}}}
 \end{array}$$

□

The reduction of $i\downarrow$ and $i\uparrow$ to $ai\downarrow$ and $ai\uparrow$ through $g\downarrow$ and $g\uparrow$ is a first successful test of the internal symmetries of the system we are building. The ultimate test will be of course cut elimination.

The derivability of cut for a system containing an atomic cut is peculiar to our calculus of structures (one usually has just admissibility of cut, no matter whether atomic or not). For example, there is no such property in the sequent system of linear logic, while the same system enjoys the derivability of identity for a system containing the atomic identity. The reason of this is very simple: the polarity of the inference rules of the sequent calculus (introduction of the main connective going down in a derivation) goes in the right direction for the factorisation of identity, and in the wrong one for cut. The reduction to atomic cuts does take place in linear logic, but only when eliminating cuts in proofs. After performing appropriate permutations of rule instances in a proof, this is a typical reduction that could occur:

$$\begin{array}{c}
 \begin{array}{ccc}
 \triangle \Pi & \triangle \Pi' & \triangle \Pi'' \\
 \frac{\vdash A, \Phi \quad \vdash B, \Phi'}{\otimes \frac{\vdash A \otimes B, \Phi, \Phi'}{\text{cut} \frac{\vdash \Phi, \Phi', \Psi}{\vdash \Phi, \Phi', \Psi}}} & \frac{\vdash \bar{A}, \bar{B}, \Psi}{\wp \frac{\vdash \bar{A} \wp \bar{B}, \Psi}{\vdash \bar{A} \wp \bar{B}, \Psi}} & \\
 \end{array}
 \longrightarrow
 \begin{array}{ccc}
 \triangle \Pi & \triangle \Pi'' & \triangle \Pi' \\
 \frac{\vdash A, \Phi \quad \vdash \bar{A}, \bar{B}, \Psi}{\text{cut} \frac{\vdash \bar{B}, \Phi, \Psi}{\vdash \Phi, \Phi', \Psi}} & & \vdash B, \Phi'
 \end{array}
 \end{array}$$

Compare this situation with the second derivation shown in the proof of Theorem 3.4.3, where A and B stand for P and Q , and Φ , Φ' and Ψ constitute context $S\{\}$. A case like this, where the main connectives in eigenformulae are introduced immediately above the cut, can only be guaranteed in proofs. In our system the scope of the reduction to atomic cuts is broader: the structures introduced by a cut can *always* be introduced atom by atom by atomic cuts.

I can now get back to the question of interdependence of $g\downarrow$ and $g\uparrow$, which I posed in the preceding section: by using the interaction rules, we can see that $g\downarrow$ and $g\uparrow$ are mutually derivable.

3.4.4 Theorem *The rule $g\uparrow$ is derivable for $\{i\downarrow, i\uparrow, g\downarrow\}$; symmetrically, the rule $g\downarrow$ is derivable for $\{i\downarrow, i\uparrow, g\uparrow\}$.*

Proof Let $g\uparrow \frac{S(R, T)}{S\{Q\}}$ and $g\downarrow \frac{S\{Q\}}{S[R, T]}$ be two rule instances such that $Q \in R \diamond T$. By using

Theorem 3.1.7 we can replace them, respectively, with

$$\frac{\frac{\frac{i\downarrow \frac{S(R,T)}{S([Q,\bar{Q}],R,T)} \quad g\downarrow \frac{S([Q,(\bar{Q},R,T)]}{S(Q,([\bar{R},\bar{T}],R,T))}}{g\downarrow \frac{S(Q,([\bar{R},\bar{T}],R,T))}{S\{Q\}}} \quad \text{and} \quad \frac{\frac{\frac{i\downarrow \frac{S\langle Q \rangle}{S(Q,([\bar{R},\bar{T}],R,T))} \quad g\uparrow \frac{S(Q,([\bar{Q},R,T])}{S([Q,\bar{Q}],R,T)}}{g\uparrow \frac{S([Q,\bar{Q}],R,T)}{S[R,T]}} .$$

□

3.4.5 Corollary *The rule $q\uparrow$ is derivable for $\{i\downarrow, i\uparrow, q\downarrow, s\}$; symmetrically, the rule $q\downarrow$ is derivable for $\{i\downarrow, i\uparrow, q\uparrow, s\}$.*

Let ρ be a rule and π be its corule, i.e., π is the scheme obtained by exchanging and negating premiss and conclusion in ρ . The rule π is then derivable for the system $\{i\downarrow, i\uparrow, s, \rho\}$, because each instance $\pi \frac{S\{Q\}}{S\{P\}}$ can be replaced by

$$\frac{\frac{\frac{i\downarrow \frac{S\{Q\}}{S(Q,[P,\bar{P}])} \quad s \frac{S(P,(Q,\bar{P}))}{S[P,(Q,\bar{P})]} \quad \rho \frac{S[P,(Q,\bar{Q})]}{S\{P\}}}{i\uparrow \frac{S\{P\}}{S\{P\}}} .$$

This means, basically, that all the imaginable, future corules in the ‘up’ fragment can be reduced to the corresponding rules via the cut rule. This is not a surprise, of course, and is perfectly analogous in spirit to what happens in the sequent calculus. The reader may then wonder about the reason why I emphasise so much dealing with the ‘up’ fragment. Consider the rule $q\uparrow$: it is derivable for a system with $i\uparrow$ and $q\downarrow$, but at the same time $i\uparrow$ itself is derivable for $\{ai\uparrow, q\uparrow, s\}$. There is a *mutual* dependence between $q\uparrow$ and $i\uparrow$.

Let us put together our simplest rules and give a name to the resulting system.

3.4.6 Definition The system $\{ai\downarrow, ai\uparrow, q\downarrow, q\uparrow, s\}$, shown in Figure 10, is called *symmetric basic system* $\vee$, or SBV; $\{ai\downarrow, q\downarrow, s\}$ and $\{ai\uparrow, q\uparrow, s\}$ are its *down* and *up fragments*, respectively.

The following easy theorem provides some systems equivalent to SBV.

3.4.7 Theorem *System SBV is strongly equivalent to the systems $\{i\downarrow, i\uparrow, g\downarrow\}$, $\{i\downarrow, i\uparrow, g\uparrow\}$, $\{ai\downarrow, i\uparrow, g\downarrow\}$ and $\{i\downarrow, ai\uparrow, g\uparrow\}$.*

Proof Let us prove strong equivalence of the system $\{i\downarrow, i\uparrow, g\downarrow\}$ and SBV: $i\downarrow$ can be replaced with $\{ai\downarrow, q\downarrow, s\}$ (Theorem 3.4.2); $i\uparrow$ with $\{ai\uparrow, q\uparrow, s\}$ (Theorem 3.4.3); $g\downarrow$ with $\{q\downarrow, s\}$ (Theorem 3.2.5). Conversely, the rule $ai\downarrow$ is an instance of $i\downarrow$, the rule $ai\uparrow$ of $i\uparrow$, the rules $q\downarrow$ and s are instances of $g\downarrow$; by Corollary 3.4.5, the rule $q\uparrow$ can be replaced with $\{i\downarrow, i\uparrow, q\downarrow, s\}$, therefore with $\{i\downarrow, i\uparrow, g\downarrow\}$. Proceed analogously for $\{i\downarrow, i\uparrow, g\uparrow\}$. The system $\{ai\downarrow, i\uparrow, g\downarrow\}$ is equivalent to $\{i\downarrow, i\uparrow, g\downarrow\}$ because $ai\downarrow$ is an instance of $i\downarrow$, and by Theorem 3.4.2. Proceed analogously for $\{i\downarrow, ai\uparrow, g\uparrow\}$. □

$$\begin{array}{ccc}
\text{ai}\downarrow \frac{S\{\circ\}}{S[a, \bar{a}]} & & \text{ai}\uparrow \frac{S(a, \bar{a})}{S\{\circ\}} \\
\text{q}\downarrow \frac{S\langle [R, T]; [R', T'] \rangle}{S[\langle R; R' \rangle, \langle T; T' \rangle]} & & \text{q}\uparrow \frac{S(\langle R; T \rangle, \langle R'; T' \rangle)}{S[\langle (R, R') \rangle; \langle (T, T') \rangle]} \\
& & \text{s} \frac{S([R, T], R')}{S[(R, R'), T]}
\end{array}$$

Fig. 10 *Symmetric basic system $\vee$ (SBV)*

3.4.8 Remark Strong equivalence of SBV and $\{\text{ai}\downarrow, \text{i}\uparrow, \text{g}\downarrow\}$, so of SBV and $\{\text{ai}\downarrow, \text{i}\uparrow, \text{q}\downarrow, \text{s}\}$, is remarkable because the entire up fragment of SBV that is not common to the down one (i.e., the rules $\text{ai}\uparrow$ and $\text{q}\uparrow$) is concentrated in the rule $\text{i}\uparrow$. See Remark 4.2.4.

3.4.9 Remark All the rules of SBV enjoy an important property: a certain notion of computational cost is constant for each of them. In $\text{ai}\downarrow$ and $\text{ai}\uparrow$ we have to check whether two atoms are dual through negation and erase them, or we have to introduce two dual atoms, depending on the interpretation. In $\text{q}\downarrow$, $\text{q}\uparrow$ and s we have to rearrange some pointers to the structures involved, without actually looking at the structures (think of the graphical representation we saw). None of the rules in $\{\text{i}\downarrow, \text{i}\uparrow, \text{g}\downarrow, \text{g}\uparrow\}$ enjoys this property, because these rules involve actually inspecting the structures upon which they act.

Please note that there are no logical axioms in SBV, and derivations may grow indefinitely in both directions: $\text{ai}\downarrow$ and $\text{ai}\uparrow$ may always be used.

4 Breaking the Symmetry

In the end, is the cut rule admissible? If we just look at derivations, the cut rule itself is not replaceable with any other rule that we already have, except for the atomic cut rule, since it is the only one that produces matter, i.e., atoms. But of course we want to look at admissibility for *proofs*, i.e., special derivations in which all the matter is finally consumed.

A proof is an intrinsically asymmetric object, because at one side it has the formula to be proved, and at the other it has logical axioms—in our case emptiness. Since there are no logical axioms in SBV, we have no proofs. Let us add a logical axiom, and break the symmetry, by not adding its ‘coaxiom’. (If I did so, I would get ‘coprovability’: a structure is ‘coprovable’ when its negation is provable. Since there is no point in carrying on this symmetry when I am interested in a non-symmetric idea, I leave the symmetry broken.)

4.1 Definition The following (logical axiom) rule $\circ\downarrow$ is called *unit*:

$$\circ\downarrow \frac{}{\circ} \quad .$$

$$\begin{array}{c}
\circ\downarrow \frac{}{\circ} \quad \text{ai}\downarrow \frac{S\{\circ\}}{S[a, \bar{a}]} \quad \text{q}\downarrow \frac{S\langle[R, T]; [R', T']\rangle}{S[\langle R; R' \rangle, \langle T; T' \rangle]} \quad \text{s} \frac{S([R, T], R')}{S[(R, R'), T]}
\end{array}$$

Fig. 11 Basic system $\vee$ (BV)

A *proof* is a derivation whose topmost inference is an instance of the unit rule. Proofs are denoted by Π . A formal system $\mathcal{S}$ *proves* R if there is in $\mathcal{S}$ a proof Π whose conclusion is R , written $\Pi \Vdash_{\mathcal{S}}$ (the name Π can be omitted). Two systems are (*weakly*) *equivalent* if

R

they prove the same structures.

Please note that $\circ\downarrow$ can only occur once in a derivation, and only at the top. Of course, I could have defined provability as the possibility of observing $\circ$ on top of a derivation, without introducing $\circ\downarrow$. I prefer following the traditional way, with logical axioms, because this will allow, in future extensions, to express more complex provability observations in formal systems, rather than in definitions.

4.2 Definition An inference rule ρ is *admissible* for the formal system $\mathcal{S}$ if $\rho \notin \mathcal{S}$ and for every proof $\Vdash_{\mathcal{S} \cup \{\rho\}}$ there exists a proof $\Vdash_{\mathcal{S}}$.

Of course, derivability implies admissibility.

As a result of the break in the symmetry, the cut rule becomes superfluous for proofs, meaning that no production of matter is really necessary in order to annihilate the one we already have, if this is at all possible. It turns out that not only $i\uparrow$ (and $ai\uparrow$) is admissible, but the coseq rule $q\uparrow$ is, too: the up fragment of SBV (excluding switch, which also belongs to the down fragment) is not necessary when observing provability. What remains is the formal system BV.

4.3 Definition The system $\{\circ\downarrow, ai\downarrow, q\downarrow, s\}$, in Figure 11, is denoted BV and called *basic system $\vee$* .

This section is organised in two subsections: in the first one, the splitting theorem is proved, in the second, it is applied to proving that $q\uparrow$ and $ai\uparrow$ are admissible for BV. The splitting theorem is, perhaps as expected, technically complex. In exchange for the effort, it is also more general than ordinary cut elimination, and is applicable to other logical systems we are currently studying.

4.1 The Splitting Theorem

The classical arguments for proving cut elimination in the sequent calculus rely on the following property: when the principal formulae in a cut are active in both branches, they determine which rules are applied immediately above the cut. This is a consequence of the fact that formulae have a root connective, and logical rules only hinge on that, and nowhere else in the formula.

This property does not hold in the calculus of structures. Further, since rules can

be applied anywhere deep inside structures, everything can happen above a cut. This complicates considerably the task of proving cut elimination. On the other hand, a great simplification is made possible in the calculus of structures by the reduction of cut to its atomic form. The remaining difficulty is actually understanding what happens, while going up in a proof, *around* the atoms produced by an atomic cut. The two atoms of an atomic cut can be produced inside any structure, and they do not belong to distinct branches, as in the sequent calculus: complex interactions with their context are possible. As a consequence, our techniques are largely different from the traditional ones.

Two approaches to cut elimination in the calculus of structures have been explored in other papers: in [9] we relied on permutations of rules, in [6] the authors relied on semantics. In this paper we use a third technique, called *splitting*, which has the advantage of being more uniform than the one based on permutations and which yields a much simpler case analysis. It also establishes a deep connection to the sequent calculus, at least for the fragments of systems that allow for a sequent calculus presentation (in this case, the commutative fragment). Since many systems are expressed in the sequent calculus, our method appears to be entirely general; still it is independent of the sequent calculus and of a complete semantics.

Splitting can be best understood by considering a sequent system with no weakening and contraction. Consider for example multiplicative linear logic: If we have a proof of the sequent

$$\vdash F\{A \otimes B\}, \Gamma \quad ,$$

where $F\{A \otimes B\}$ is a formula that contains the subformula $A \otimes B$, we know for sure that somewhere in the proof there is one and only one instance of the $\otimes$ rule that splits A and B along with their context. We are in the following situation:

$$\begin{array}{ccc}
 \begin{array}{c}
 \begin{array}{c} \Pi_1 \quad \Pi_2 \\ \hline \vdash A, \Phi \quad \vdash B, \Psi \\ \hline \otimes \quad \vdash A \otimes B, \Phi, \Psi \end{array} \\
 \Delta \\
 \hline
 \vdash F\{A \otimes B\}, \Gamma
 \end{array}
 & \text{corresponds to} &
 \begin{array}{c}
 \Pi_2 \top \\
 [B, \Psi] \\
 \Pi_1 \parallel \\
 \frac{([A, \Phi], [B, \Psi])}{s} \\
 \frac{[[A, \Phi], B), \Psi]}{s} \\
 [(A, B), \Phi, \Psi] \\
 \Delta \parallel \\
 [F(A, B), \Gamma]
 \end{array}
 \end{array}$$

We can consider, as shown at the left, the proof for the given sequent as composed of three pieces, Δ , Π_1 and Π_2 . In the calculus of structures, many different proofs correspond to the sequent calculus one: they differ for the different possible sequencing of rules, and because rules in the calculus of structures have smaller granularity and larger applicability. But, among all these proofs, there must also be one that fits the scheme at the right of the figure above. This precisely illustrates the idea behind the splitting technique.

The derivation Δ above implements a *context reduction* and a proper splitting. We can state, in general, these principles as follows:

- 1 Context reduction: If $S\{R\}$ is provable, then $S\{ \}$ can be reduced, going up in a proof, to the structure $[\{ \}, U]$, such that $[R, U]$ is provable. In the example above, $[F\{ \}, \Gamma]$ is reduced to $[\{ \}, \Gamma']$, for some Γ' .
- 2 (Shallow) splitting: If $[(R, T), P]$ is provable, then P can be reduced, going up in a proof, to $[P_1, P_2]$, such that $[R, P_1]$ and $[T, P_2]$ are provable. In the example above Γ' is reduced to $[\Phi, \Psi]$.

Context reduction is in turn proved by splitting, which is then at the core of the matter. The biggest difficulty resides in proving splitting, and this mainly requires finding the right induction measure. We first need to establish two easy propositions, which express obvious properties induced by the rules of BV. I will often use them implicitly throughout the rest of the paper.

4.1.1 Proposition *The size of the premiss of a derivation in BV is not greater than the size of the conclusion.*

4.1.2 Proposition *In BV, $\langle R; T \rangle$ is provable if and only if R and T are provable and (R, T) is provable if and only if R and T are provable.*

The calculus of structures is of course very similar to a term rewriting system, and especially so for those, like me, who have an intuitive bias toward considering derivations growing in a bottom-up fashion. In this case the direction in which rewriting occurs is clear, and it is natural to use some terminology of rewriting systems for inference rules.

4.1.3 Definition The inference rules of SBV are all of the kind $\rho \frac{S\{V\}}{S\{U\}}$: the structure U is called the *redex* (and V is the *contractum*) of the rule's instance.

4.1.4 Theorem (Shallow Splitting) *For all structures R, T and P :*

- 1 *if $[\langle R; T \rangle, P]$ is provable in BV then there exist P_1, P_2 and $\frac{\langle P_1; P_2 \rangle}{P} \in \text{BV}$ such that $[R, P_1]$ and $[T, P_2]$ are provable in BV;*
- 2 *if $[(R, T), P]$ is provable in BV then there exist P_1, P_2 and $\frac{[P_1, P_2]}{P} \in \text{BV}$ such that $[R, P_1]$ and $[T, P_2]$ are provable in BV.*

Proof All the derivations appearing in this argument are in BV. Let us consider the lexicographic order $\prec$ on natural numbers defined by $(m', n') \prec (m, n)$ iff either $m' < m$, or $m' = m$ and $n' < n$.

Consider the following two statements:

$$\begin{aligned}
S(m, n) = & \forall m', n'. \forall R, T, P. \left(((m', n') \preceq (m, n) \right. \\
& \wedge m' = |\text{occ} [\langle R; T \rangle, P]| \\
& \wedge \text{there is a proof } \prod \text{ of length } n') \\
& \quad \left. [\langle R; T \rangle, P] \right. \\
& \quad \left. \langle P_1; P_2 \rangle \right. \\
& \Rightarrow \exists P_1, P_2. \left(\prod_P \wedge \prod_{[R, P_1]} \wedge \prod_{[T, P_2]} \right) \quad , \\
C(m, n) = & \forall m', n'. \forall R, T, P. \left(((m', n') \preceq (m, n) \right. \\
& \wedge m' = |\text{occ} [(R, T), P]| \\
& \wedge \text{there is a proof } \prod \text{ of length } n') \\
& \quad \left. [(R, T), P] \right. \\
& \quad \left. [P_1, P_2] \right. \\
& \Rightarrow \exists P_1, P_2. \left(\prod_P \wedge \prod_{[R, P_1]} \wedge \prod_{[T, P_2]} \right) \quad .
\end{aligned}$$

The statement of the theorem is equivalent to $\forall m, n. (S(m, n) \wedge C(m, n))$. We can consider (m, n) a measure of $(S(m, n) \wedge C(m, n))$, and the proof is an induction, by $\prec$, on this measure. The base case being trivial, let us see the inductive ones. We will always assume $P \neq \circ$, since when $P = \circ$ the theorem is trivially proved by Proposition 4.1.2. For the same reason, we assume $R \neq \circ \neq T$. We will prove separately $S(m, n)$ and $C(m, n)$.

- 1** $\forall m', n'. ((m', n') \prec (m, n) \wedge S(m', n') \wedge C(m', n')) \Rightarrow S(m, n)$: The size of $[\langle R; T \rangle, P]$ is m and there is a proof of it of length n . Let us consider the bottom rule instance in this proof:

$$\rho \frac{\prod_Q}{[\langle R; T \rangle, P]} \quad ,$$

where we assume that ρ is non-trivial, otherwise the induction hypothesis applies. Let us reason on the position of the redex of ρ in $[\langle R; T \rangle, P]$. Here are the possibilities:

- 1** $\rho = \text{ai}\downarrow$: The following cases exhaust the possibilities:

- 1** The redex is inside R :

$$\text{given } \text{ai}\downarrow \frac{\prod_{[\langle R'; T \rangle, P]}}{[\langle R; T \rangle, P]} \quad , \quad \text{consider } \text{ai}\downarrow \frac{\prod_{[R', P_1]}}{[R, P_1]} \quad .$$

- 2** The redex is inside T : Analogous to the previous case.

- 3** The redex is inside P :

$$\text{given } \text{ai}\downarrow \frac{\prod_{[\langle R; T \rangle, P']}}{[\langle R; T \rangle, P]} \quad , \quad \text{consider } \text{ai}\downarrow \frac{\prod_{P'}}{P} \quad .$$

- 2** $\rho = \text{q}\downarrow$: If the redex is inside R , T or P , we have analogous situations to the ones seen above. The following cases exhaust the other possibilities:

- 1** $R = \langle R'; R'' \rangle$, $P = [\langle P'; P'' \rangle, U]$ and

$$\text{q}\downarrow \frac{\begin{array}{c} \parallel \\ \frac{[\langle [R', P']; [\langle R''; T \rangle, P''] \rangle, U]}{[\langle R'; R''; T \rangle, \langle P'; P'' \rangle, U]} \end{array}}{\quad} :$$

We can apply the induction hypothesis, and we get:

$$\begin{array}{c} \langle U_1; U_2 \rangle \\ \parallel \\ U \end{array}, \quad \begin{array}{c} \parallel \\ [R', P', U_1] \end{array} \quad \text{and} \quad \begin{array}{c} \pi \parallel \\ [\langle R''; T \rangle, P'', U_2] \end{array} .$$

Since $|\text{occ} [\langle R''; T \rangle, P'', U_2]| < |\text{occ} [\langle R'; R''; T \rangle, \langle P'; P'' \rangle, U]|$ (otherwise the $\text{q}\downarrow$ instance would be trivial), we can apply the induction hypothesis on Π and get:

$$\begin{array}{c} \langle P'_1; P_2 \rangle \\ \parallel \\ [P'', U_2] \end{array}, \quad \begin{array}{c} \parallel \\ [R'', P'_1] \end{array} \quad \text{and} \quad \begin{array}{c} \parallel \\ [T, P_2] \end{array} .$$

We can now take $P_1 = \langle [P', U_1]; P'_1 \rangle$ and build

$$\begin{array}{c} \langle [P', U_1]; P'_1; P_2 \rangle \\ \parallel \\ \text{q}\downarrow \frac{\langle [P', U_1]; [P'', U_2] \rangle}{[\langle P'; P'' \rangle, \langle U_1; U_2 \rangle]} \end{array} \quad \begin{array}{c} \parallel \\ \text{q}\downarrow \frac{\langle [R', P', U_1]; [R'', P'_1] \rangle}{[\langle R'; R'' \rangle, \langle [P', U_1]; P'_1 \rangle]} \end{array} .$$

A similar argument holds when $T = \langle T'; T'' \rangle$ and we are given a proof

$$\text{q}\downarrow \frac{\begin{array}{c} \parallel \\ \frac{[\langle [R; T'], P' \rangle; [T'', P''] \rangle, U]}{[\langle R; T'; T'' \rangle, \langle P'; P'' \rangle, U]} \end{array}}{\quad} .$$

- 2** $P = [\langle P'; P'' \rangle, U', U'']$ and

$$\text{q}\downarrow \frac{\begin{array}{c} \parallel \\ \frac{[\langle [R; T], P', U' \rangle; P''] \rangle, U'']}{[\langle R; T \rangle, \langle P'; P'' \rangle, U', U'']} \end{array}}{\quad} :$$

We can apply the induction hypothesis, and we get:

$$\begin{array}{c} \langle U_1; U_2 \rangle \\ \parallel \\ U'' \end{array}, \quad \begin{array}{c} \pi \parallel \\ [\langle R; T \rangle, P', U', U_1] \end{array} \quad \text{and} \quad \begin{array}{c} \parallel \\ [P'', U_2] \end{array} .$$

Since $|\text{occ} [\langle R; T \rangle, P', U', U_1]| < |\text{occ} [\langle R; T \rangle, \langle P'; P'' \rangle, U', U'']|$ (otherwise the $\text{q}\downarrow$ instance would be trivial), we can apply the induction hypothesis on Π and get:

$$\begin{array}{c} \langle P_1; P_2 \rangle \\ \parallel \\ [P', U', U_1] \end{array}, \quad \begin{array}{c} \prod \\ [R, P_1] \end{array} \quad \text{and} \quad \begin{array}{c} \prod \\ [T, P_2] \end{array}.$$

We can now build

$$\begin{array}{c} \langle P_1; P_2 \rangle \\ \parallel \\ [P', U_1, U'] \\ \parallel \\ \text{q}\downarrow \frac{[\langle [P', U_1]; [P'', U_2] \rangle, U']}{[\langle P'; P'' \rangle, U', \langle U_1; U_2 \rangle]} \\ \parallel \\ [\langle P'; P'' \rangle, U', U''] \end{array}.$$

A similar argument holds when we are given a proof

$$\begin{array}{c} \prod \\ \text{q}\downarrow \frac{[\langle P'; [\langle R; T \rangle, P'', U'] \rangle, U'']}{[\langle R; T \rangle, \langle P'; P'' \rangle, U', U'']} \end{array}.$$

- 3** $\rho = \text{s}$: If the redex is inside R , T or P , we have analogous situations to the ones seen in Case 1.1. The only other possibility is the following; let $P = [(P', P''), U', U'']$ and the given proof is

$$\begin{array}{c} \prod \\ \text{s} \frac{[\langle [\langle R; T \rangle, P', U'], P'' \rangle, U'']}{[\langle R; T \rangle, \langle P'; P'' \rangle, U', U'']} \end{array}.$$

We can apply the induction hypothesis, and we get:

$$\begin{array}{c} [U_1, U_2] \\ \parallel \\ U'' \end{array}, \quad \begin{array}{c} \prod \\ [\langle R; T \rangle, P', U', U_1] \end{array} \quad \text{and} \quad \begin{array}{c} \prod \\ [P'', U_2] \end{array}.$$

Since $|\text{occ} [\langle R; T \rangle, P', U', U_1]| < |\text{occ} [\langle R; T \rangle, \langle P'; P'' \rangle, U', U'']|$ (otherwise the s instance would be trivial), we can apply the induction hypothesis on Π and get:

$$\begin{array}{c} \langle P_1; P_2 \rangle \\ \parallel \\ [P', U', U_1] \end{array}, \quad \begin{array}{c} \prod \\ [R, P_1] \end{array} \quad \text{and} \quad \begin{array}{c} \prod \\ [T, P_2] \end{array}.$$

We can now build

$$\begin{array}{c}
 \langle P_1; P_2 \rangle \\
 \parallel \\
 [P', U', U_1] \\
 \parallel \\
 \text{s} \frac{[(P', [P'', U_2]), U', U_1]}{[(P', P''), U', U_1, U_2]} \\
 \parallel \\
 [(P', P''), U', U'']
 \end{array}$$

- 2** $\forall m', n'. ((m', n') \prec (m, n) \wedge S(m', n') \wedge C(m', n')) \Rightarrow C(m, n)$: The size of $[(R, T), P]$ is m and there is a proof of it of length n . Let us consider the bottom rule instance in this proof:

$$\begin{array}{c}
 \parallel \\
 \text{Q} \\
 \rho \frac{}{[(R, T), P]}
 \end{array}
 ,$$

where we assume that ρ is non-trivial, otherwise the induction hypothesis applies. Let us reason on the position of the redex of ρ in $[(R, T), P]$. Here are the possibilities:

- 1** $\rho = \text{ai}\downarrow$: Analogous to Case 1.1.
2 $\rho = \text{q}\downarrow$: If the redex is inside R , T or P , we have analogous situations to the ones seen in Case 1.1. The only other possibility is the following; let $P = [\langle P'; P'' \rangle, U', U'']$ and the given proof is

$$\begin{array}{c}
 \parallel \\
 \text{q}\downarrow \frac{[\langle [(R, T), P', U']; P'' \rangle, U'']}{[(R, T), \langle P'; P'' \rangle, U', U'']}
 \end{array}$$

We can apply the induction hypothesis, and we get:

$$\begin{array}{c}
 \langle U_1; U_2 \rangle \\
 \parallel \\
 U''
 \end{array}
 , \quad
 \begin{array}{c}
 \parallel \\
 [(R, T), P', U', U_1]
 \end{array}
 \quad \text{and} \quad
 \begin{array}{c}
 \parallel \\
 [P'', U_2]
 \end{array}
 .$$

Since $|\text{occ} [(R, T), P', U', U_1]| < |\text{occ} [(R, T), \langle P'; P'' \rangle, U', U'']|$ (otherwise the $\text{q}\downarrow$ instance would be trivial), we can apply the induction hypothesis on Π and get:

$$\begin{array}{c}
 [P_1, P_2] \\
 \parallel \\
 [P', U', U_1]
 \end{array}
 , \quad
 \begin{array}{c}
 \parallel \\
 [R, P_1]
 \end{array}
 \quad \text{and} \quad
 \begin{array}{c}
 \parallel \\
 [T, P_2]
 \end{array}
 .$$

We can now build

$$\begin{array}{c}
 [P_1, P_2] \\
 \parallel \\
 [P', U', U_1] \\
 \parallel \\
 \text{q}\downarrow \frac{[\langle [P', U_1]; [P'', U_2] \rangle, U']}{[\langle P'; P'' \rangle, U', \langle U_1; U_2 \rangle]} \\
 \parallel \\
 [\langle P'; P'' \rangle, U', U'']
 \end{array}
 .$$

A similar argument holds when we are given a proof

$$\begin{array}{c} \top \\ \text{q}\downarrow \frac{[\langle P'; [(R, T), P''], U']\rangle, U'']}{[(R, T), \langle P'; P''\rangle, U', U'']} \end{array} .$$

3 $\rho = s$: If the redex is inside R , T or P , we have analogous situations to the ones seen in Case 1.1. The following cases exhaust the other possibilities:

1 $R = (R', R'')$, $T = (T', T'')$, $P = [P', P'']$ and

$$\begin{array}{c} \top \\ s \frac{[\langle [(R', T'), P'], R'', T''\rangle, P'']}{[(R', R'', T', T''), P', P'']} \end{array} :$$

We can apply the induction hypothesis, and we get:

$$\begin{array}{c} [P'_1, P'_2] \\ \parallel \\ P'' \end{array}, \quad \begin{array}{c} \Pi \\ \parallel \\ [(R', T'), P', P'_1] \end{array} \quad \text{and} \quad \begin{array}{c} \Pi' \\ \parallel \\ [(R'', T''), P'_2] \end{array} .$$

$|\text{occ} [(R', T'), P', P'_1]| < |\text{occ} [(R', R'', T', T''), P', P'']| > |\text{occ} [(R'', T''), P'_2]|$ holds (otherwise the s instance would be trivial), then we can apply twice the induction hypothesis on Π and Π' and get:

$$\begin{array}{c} [P'_1, P'_2] \\ \parallel \\ [P', P'_1] \end{array}, \quad \begin{array}{c} \top \\ \parallel \\ [R', P'_1] \end{array} \quad \text{and} \quad \begin{array}{c} \top \\ \parallel \\ [T', P'_2] \end{array};$$

$$\begin{array}{c} [P''_1, P''_2] \\ \parallel \\ P'_2 \end{array}, \quad \begin{array}{c} \top \\ \parallel \\ [R'', P''_1] \end{array} \quad \text{and} \quad \begin{array}{c} \top \\ \parallel \\ [T'', P''_2] \end{array} .$$

We can now take $P_1 = [P'_1, P''_1]$, $P_2 = [P'_2, P''_2]$ and build

$$\begin{array}{c} [P'_1, P'_2, P''_1, P''_2] \\ \parallel \\ [P'_1, P'_2, P'_2] \end{array} \quad \begin{array}{c} \top \\ \parallel \\ [R'', P''_1] \end{array} \quad \begin{array}{c} \top \\ \parallel \\ [T'', P''_2] \end{array}$$

$$\begin{array}{c} \parallel \\ [P', P'_1, P'_2] \end{array} \quad \begin{array}{c} \parallel \\ [P', P''] \end{array}, \quad s \frac{[\langle [(R', P'_1), R''], P''_1] \rangle, P''_1]}{[(R', R''), P'_1, P''_1]} \quad \text{and} \quad s \frac{[\langle [(T', P'_2), T''], P''_2] \rangle, P''_2]}{[(T', T''), P'_2, P''_2]} .$$

2 $P = [(P', P''), U', U'']$ and

$$\begin{array}{c} \top \\ s \frac{[\langle [(R, T), P', U'], P''\rangle, U'']}{[(R, T), \langle P', P''\rangle, U', U'']} \end{array} :$$

We can apply the induction hypothesis, and we get:

$$\begin{array}{c} [U_1, U_2] \\ \parallel \\ U'' \end{array}, \quad \begin{array}{c} \Pi \\ [(R, T), P', U', U_1] \end{array} \quad \text{and} \quad \begin{array}{c} \Pi \\ [P'', U_2] \end{array}.$$

Since $|\text{occ}[(R, T), P', U', U_1]| < |\text{occ}[(R, T), (P', P''), U', U'']|$ (otherwise the instance would be trivial), we can apply the induction hypothesis on Π and get:

$$\begin{array}{c} [P_1, P_2] \\ \parallel \\ [P', U', U_1] \end{array}, \quad \begin{array}{c} \Pi \\ [R, P_1] \end{array} \quad \text{and} \quad \begin{array}{c} \Pi \\ [T, P_2] \end{array}.$$

We can now build

$$\begin{array}{c} [P_1, P_2] \\ \parallel \\ [P', U', U_1] \\ \parallel \\ \frac{[(P', [P'', U_2]), U', U_1]}{[(P', P''), U', U_1, U_2]} \text{ s} \\ \parallel \\ [(P', P''), U', U''] \end{array}.$$

□

4.1.5 Theorem (Context Reduction) *For all structures R and for all contexts $S\{\ \}$ such that $S\{R\}$ is provable in BV , there exists a structure U such that for all structures X there exist derivations:*

$$\begin{array}{c} [X, U] \\ \parallel_{\text{BV}} \\ S\{X\} \end{array} \quad \text{and} \quad \begin{array}{c} \Pi_{\text{BV}} \\ [R, U] \end{array}.$$

Proof By induction on the size of $S\{\circ\}$. The base case is trivial: $U = \circ$. There are three inductive cases:

- 1 $S\{\ \} = \langle S'\{\ \}; P \rangle$, for some $P \neq \circ$: By Proposition 4.1.2 there are proofs in BV of $S'\{R\}$ and of P . By applying the induction hypothesis, we can find U and build, for all X :

$$\begin{array}{c} [X, U] \\ \parallel_{\text{BV}} \\ S'\{X\} \\ \parallel_{\text{BV}} \\ \langle S'\{X\}; P \rangle \end{array},$$

such that $[R, U]$ is provable in BV . The same argument applies when $S\{\ \} = \langle P; S'\{\ \} \rangle$, for some $P \neq \circ$.

- 2 $S\{\ \} = [S'\{\ \}, P]$, for some $P \neq \circ$ such that $S'\{\ \}$ is not a proper par: If $S'\{\circ\} = \circ$ then the theorem is proved; otherwise there are two possibilities:

- 1** $S'\{ \} = \langle S''\{ \}; P' \rangle$, for some $P' \neq \circ$: By Theorem 4.1.4 there exist:

$$\begin{array}{c} \langle P_1; P_2 \rangle \\ \parallel_{\text{BV}} \\ P \end{array}, \quad \begin{array}{c} \Pi \\ \parallel_{\text{BV}} \\ [S''\{R\}, P_1] \end{array} \quad \text{and} \quad \begin{array}{c} \parallel_{\text{BV}} \\ [P', P_2] \end{array}.$$

By applying the induction hypothesis to Π , we get:

$$\begin{array}{c} [X, U] \\ \parallel_{\text{BV}} \\ [S''\{X\}, P_1] \\ \parallel_{\text{BV}} \\ \text{q}\downarrow \frac{\langle [S''\{X\}, P_1]; [P', P_2] \rangle}{[\langle S''\{X\}; P' \rangle, \langle P_1; P_2 \rangle]} \end{array} \quad \text{and} \quad \begin{array}{c} \parallel_{\text{BV}} \\ [R, U] \end{array}.$$

$$\begin{array}{c} \parallel_{\text{BV}} \\ [\langle S''\{X\}; P' \rangle, P] \end{array}$$

We can proceed analogously when $S'\{ \} = \langle P'; S''\{ \} \rangle$, for some $P' \neq \circ$.

- 2** $S'\{ \} = (S''\{ \}, P')$, for some $P' \neq \circ$: Analogous to the previous case.
- 3** $S\{ \} = (S'\{ \}, P)$, for some $P \neq \circ$: Analogous to Case 1. □

4.1.6 Corollary (Splitting) *For all structures R, T and for all contexts $S\{ \}$:*

- 1** *if $S\langle R; T \rangle$ is provable in BV then there exist structures S_1 and S_2 such that, for all structures X , there exists a derivation*

$$\begin{array}{c} [X, \langle S_1; S_2 \rangle] \\ \parallel_{\text{BV}} \\ S\{X\} \end{array};$$

- 2** *if $S(R, T)$ is provable in BV then there exist structures S_1 and S_2 such that, for all structures X , there exists a derivation*

$$\begin{array}{c} [X, S_1, S_2] \\ \parallel_{\text{BV}} \\ S\{X\} \end{array};$$

and, in both cases, there are proofs $\parallel_{\text{BV}}$ and $\parallel_{\text{BV}}$.

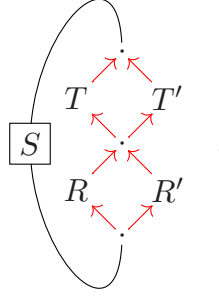
$$\begin{array}{c} [R, S_1] \\ [T, S_2] \end{array}$$

Proof First apply Theorem 4.1.5 and then apply Theorem 4.1.4. □

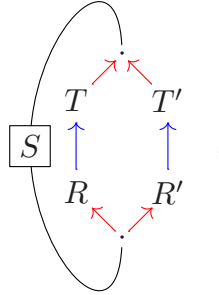
4.2 Admissibility of the Up Fragment

Let us start with admissibility of $\text{q}\uparrow$. Since I broke the symmetry, I can now informally consider interaction a phenomenon that only occurs inside par contexts while going upward

in a derivation. Take the conclusion of a $q\uparrow$ instance:



At the end of the process of rewriting this structure toward the unit axiom, the structures R , R' , T and T' must vanish, i.e., all the atoms inside them have to participate in $ai\downarrow$ interactions. Consider R : where are atoms that can annihilate its atoms? There are two possibilities: inside R itself or inside $S\{ \}$, and nowhere else. Now, we have to ask ourselves whether the application of $q\uparrow$ to the conclusion above contributes in an essential way to the vanishing of R . Through $q\uparrow$ we get



The application of $q\uparrow$ just introduces new constraints: we cannot now use a structure $\langle R''; T'' \rangle$ in $S\{ \}$ to obtain $[R, R'']$ and $[T', T'']$ as we could before. Since the same argument holds for R' , T and T' , we can conclude that, as far as provability is concerned, $q\uparrow$ should be disposable.

This argument, which is essentially, but not entirely, correct (does the reader see the problem?), seems reasonable and amenable to a direct implementation in an admissibility proof. It is in fact what I did in the beginning, through the study of permutabilities inside BV. Unfortunately, in certain special cases, the argument requires an unbearable complexity in the case analysis. The following proof uses splitting and is instead much more direct, and, in my opinion, more illuminating.

4.2.1 Theorem *The rule $q\uparrow$ is admissible for BV.*

Proof Consider the proof

$$q\uparrow \frac{\prod_{BV} S(\langle R; T \rangle, \langle R'; T' \rangle)}{S(\langle (R, R') \rangle; \langle (T, T') \rangle)} .$$

By Corollary 4.1.6 there exist S_1 and S_2 such that there are derivations:

$$\begin{array}{c} \prod_{BV} \\ S(\langle (R, R') \rangle; \langle (T, T') \rangle) \end{array}, \quad \begin{array}{c} \prod_{BV} \\ \langle R; T \rangle, S_1 \end{array} \quad \text{and} \quad \begin{array}{c} \prod_{BV} \\ \langle R'; T' \rangle, S_2 \end{array} .$$

By Theorem 4.1.4 we have, for some $S_R, S_T, S_{R'}, S_{T'}$:

$$\frac{\langle S_R; S_T \rangle}{S_1} \parallel_{\text{BV}}, \quad \frac{\parallel_{\text{BV}}}{[R, S_R]}, \quad \frac{\parallel_{\text{BV}}}{[T, S_T]}, \quad \frac{\langle S_{R'}; S_{T'} \rangle}{S_2} \parallel_{\text{BV}}, \quad \frac{\parallel_{\text{BV}}}{[R', S_{R'}]} \quad \text{and} \quad \frac{\parallel_{\text{BV}}}{[T', S_{T'}]}.$$

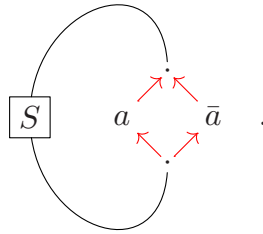
We can then build

$$\begin{array}{c} \parallel_{\text{BV}} \\ \text{q}\downarrow \frac{\langle [R', S_{R'}]; [T', S_{T'}] \rangle}{[\langle R'; T' \rangle, \langle S_{R'}; S_{T'} \rangle]} \\ \parallel_{\text{BV}} \\ \text{s} \frac{[\langle R'; ([T, S_T], T') \rangle, \langle S_{R'}; S_{T'} \rangle]}{[\langle R'; [(T, T'), S_T] \rangle, \langle S_{R'}; S_{T'} \rangle]} \\ \parallel_{\text{BV}} \\ \text{s} \frac{[\langle ([R, S_R], R'); [(T, T'), S_T] \rangle, \langle S_{R'}; S_{T'} \rangle]}{[\langle [(R, R'), S_R]; [(T, T'), S_T] \rangle, \langle S_{R'}; S_{T'} \rangle]} \\ \text{q}\downarrow \frac{[\langle (R, R'); (T, T') \rangle, \langle S_R; S_T \rangle, \langle S_{R'}; S_{T'} \rangle]}{[\langle (R, R'); (T, T') \rangle, \langle S_R; S_T \rangle, \langle S_{R'}; S_{T'} \rangle]} \\ \parallel_{\text{BV}} \\ S\langle (R, R'); (T, T') \rangle \end{array},$$

which is a proof in BV.

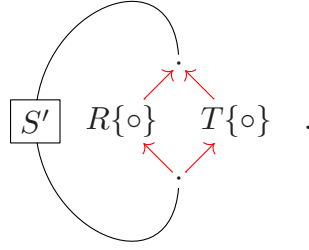
We can repeat inductively this argument, starting from the top, for any proof in $\text{BV} \cup \{\text{q}\uparrow\}$, eliminating all the $\text{q}\uparrow$ instances one by one. $\square$

Let us now consider a structure $S\{\circ\}$ that, at a certain point while going up building a proof in $\text{BV} \cup \{\text{ai}\uparrow\}$, creates two complementary atoms a and $\bar{a}$, by a cut:



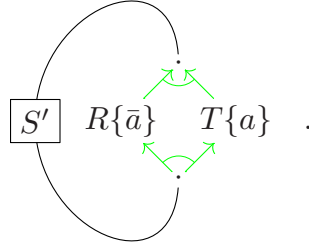
Suppose that this is the last creation of atoms, i.e., the proof of $S(a, \bar{a})$ is in BV. The atoms a and $\bar{a}$ do not communicate; therefore, since they will eventually be annihilated, there must be two other atoms $\bar{a}$ and a that will do the job of killing them. Proceeding toward the top, nothing can leave the copar context that hosts the newly created atoms a and $\bar{a}$. This means that there are, up in the proof, two structures $R\{\bar{a}\}$ and $T\{a\}$ that, at some point going upward, enter the copar, carrying the killer atoms with them. So, essentially, we have to consider the case where two structures $R\{\circ\}$ and $T\{\circ\}$ are to be

proved in this situation:

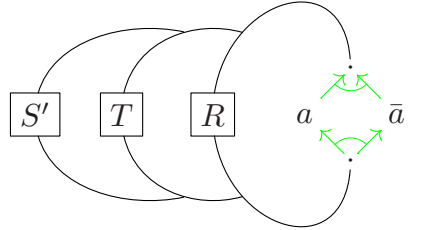


Since $R\{\circ\}$ and $T\{\circ\}$ do not communicate, all the resources needed to annihilate them must come either from inside themselves or from $S'\{\}$.

Consider now the alternative situation in which the original atoms a and $\bar{a}$ have not been created at all. Since two structures $R\{\bar{a}\}$ and $T\{a\}$ are inside $S\{\}$ and can both enter the same copar, they must live in a par structural relation. Therefore, from $S\{\circ\}$, we can reach the structure:



In fact, $R\{\bar{a}\}$ and $T\{a\}$ could enter the original copar context $(a, \bar{a})$ only by distinct uses of the rule s . If one replaces $(a, \bar{a})$ with $\circ$ and does not apply s , then $R\{\bar{a}\}$ and $T\{a\}$ remain in the same par context. But then one of $R\{\bar{a}\}$ and $T\{a\}$ can be merged into the other until $\bar{a}$ and a come into direct communication, for example:



Now the two complementary atoms a and $\bar{a}$ can be eliminated by an application of $\mathbf{ai}\downarrow$. Since $S'(R\{\circ\}, T\{\circ\})$ was provable, then $R\{\circ\}$ and $T\{\circ\}$ can be annihilated, each of them by use of its own resources, or of those coming from $S'\{\}$. This shows that the original cut was superfluous, and we can continue applying this argument on the other cut instances.

We can implement this idea in the following theorem, but before that I have to state an easy proposition.

4.2.2 Proposition *For every context $S\{\}$ and structures R, T , there exists a derivation*

$$\frac{S[R, T]}{[S\{R\}, T]} \parallel \{\mathbf{q}\downarrow, \mathbf{s}\} .$$

4.2.3 Theorem *The rule $\text{ai}\uparrow$ is admissible for BV.*

Proof Consider the proof

$$\text{ai}\uparrow \frac{\prod_{\text{BV}} S(a, \bar{a})}{S\{\circ\}}.$$

By Corollary 4.1.6 there exist S_1 and S_2 such that there are derivations:

$$\begin{array}{c} [S_1, S_2] \\ \prod_{\text{BV}} \\ S\{\circ\} \end{array}, \quad \begin{array}{c} \Pi_1 \prod_{\text{BV}} \\ [a, S_1] \end{array} \quad \text{and} \quad \begin{array}{c} \Pi_2 \prod_{\text{BV}} \\ [\bar{a}, S_2] \end{array}.$$

Consider the proof Π_1 : there must be a context $S'_1\{\ \}$ such that $S_1 = S'_1\{\bar{a}\}$ and

$$\Pi_1 = \frac{\prod_{\text{BV}} \frac{\text{ai}\downarrow \frac{S''\{\circ\}}{S''[a, \bar{a}]}}{[a, S'_1\{\bar{a}\}]},$$

for some $S''\{\ \}$, in which we individuate the instance of $\text{ai}\downarrow$ where the a occurrence interacts with the $\bar{a}$ occurrence coming from $S'_1\{\bar{a}\}$. We can replace in Π_1 every occurrence of a and $\bar{a}$ with $\circ$, and we obtain a proof in BV of $S'_1\{\circ\}$. Analogously, we can transform Π_2 into a proof in BV of an $S'_2\{\circ\}$ such that $S_2 = S'_2\{a\}$. We can then build the following proof

$$\frac{\prod_{\text{BV}} \frac{\prod_{\text{BV}} \frac{\text{ai}\downarrow \frac{S'_1\{S'_2\{\circ\}\}}{S'_1\{S'_2[a, \bar{a}]\}}}{[S'_1\{\bar{a}\}, S'_2\{a\}]}}{S\{\circ\}}},$$

where we used twice Proposition 4.2.2.

We can repeat inductively this argument, starting from the top, for any proof in $\text{BV} \cup \{\text{ai}\uparrow\}$, eliminating all the $\text{ai}\uparrow$ instances one by one. $\square$

4.2.4 Remark The admissible fragment of $\text{SBV} \cup \{\circ\downarrow\}$ is $\{\text{ai}\uparrow, \text{q}\uparrow\}$. By Theorems 3.4.7 and 3.2.5, system $\text{SBV} \cup \{\circ\downarrow\}$ is strongly equivalent to the system $\{\circ\downarrow, \text{ai}\downarrow, \text{i}\uparrow, \text{q}\downarrow, \text{s}\} = \text{BV} \cup \{\text{i}\uparrow\}$.

4.2.5 Theorem *$\text{SBV} \cup \{\circ\downarrow\}$ and BV are equivalent.*

Proof Given a proof in $\text{SBV} \cup \{\circ\downarrow\}$, eliminate the $\text{q}\uparrow$ and $\text{ai}\uparrow$ instances one by one starting from the top. $\square$

Since any atom a cannot be proved in BV, then $\text{SBV} \cup \{\circ\downarrow\}$ and BV are consistent. This consistency property can be reinforced as in the following theorem, whose proof is particularly interesting:

4.2.6 Theorem *If S is provable in BV then $\bar{S}$ is not provable, provided $S \neq \circ$.*

Proof A proof of S must be as

$$\frac{\frac{\circ \downarrow \frac{\quad}{\circ}}{\text{ai} \downarrow \frac{\quad}{[a, \bar{a}]}}}{\left\|_{\text{BV}} \frac{\quad}{S}\right.}.$$

Get $\left\|_{\text{SBV}} \frac{\bar{S}}{(a, \bar{a})}\right.$ by flipping the given proof. If $\bar{S}$ is provable, then $(a, \bar{a})$ is provable in $\text{SBV} \cup \{\circ \downarrow\}$ and then in BV: impossible. $\square$

4.2.7 Theorem *It is decidable whether a given structure is provable in $\text{SBV} \cup \{\circ \downarrow\}$ and in BV.*

Proof The set of derivations in BV that have a given conclusion, for which to look, is finite (if no trivial instances of rules are applied, which is not a limitation); see Subsection 3.3. $\square$

Most cut-free sequent systems enjoy the *subformula property*: the premisses of rule instances only contain subformulae of formulae in their conclusions. Our case is different, because we do not use connectives to split formulae in conclusions into parts that go into premisses, but it is analogous. Apart from the cut rule, all the other rules lower the ‘energy’ of a formula (see Subsection 3.3), either by the property mentioned in Remark 3.3.5 (with $\text{q} \downarrow$, $\text{q} \uparrow$ and s) or because atom occurrences diminish in number (with $\text{ai} \downarrow$). In this sense, we get simpler structures while going up in the building of a derivation.

A consequence of the subformula property is that systems are modular with respect to connectives: Given a system, one can remove a connective and its dual through negation and their relative rules, without affecting the provability of the formulae that do not use those connectives. A similar situation occurs in BV. Consider not to employ the seq structural relation, i.e., just use flat structures. For them, the following system is appropriate:

4.2.8 Definition System FBV (called *flat system BV*) is the system $\{\circ \downarrow, \text{ai} \downarrow, \text{s}\}$.

Deducing over flat structures with system FBV means remaining in a flat horizontal line where all the structural relations are commutative.

4.2.9 Theorem *If a flat structure is provable in BV then it is provable in FBV.*

Proof Let Π be a proof in BV of a given flat structure S . Transform Π into an object Π' , by changing every seq context appearing in each structure in Π into a par context. Π' is still a proof, where all the instances of $\text{q} \downarrow$ become trivial and can be eliminated, this way obtaining a proof of S in FBV. $\square$

This is a key result in the correspondence between BV and linear logic, as we are going to see in the next section.

5 Relation with Linear Logic

Linear logic is, of course, the main source of inspiration of this paper. In this section I show that BV extends the multiplicative fragment of linear logic while conserving provability

$$\boxed{
\begin{array}{l}
\text{id} \frac{}{\vdash A, \bar{A}} \quad
\wp \frac{\vdash A, B, \Phi}{\vdash A \wp B, \Phi} \quad
\otimes \frac{\vdash A, \Phi \quad \vdash B, \Psi}{\vdash A \otimes B, \Phi, \Psi} \quad
\text{mix} \frac{\vdash \Phi \quad \vdash \Psi}{\vdash \Phi, \Psi}
\end{array}
}$$

Fig. 12 System MLL+mix

in its commutative fragment. The version of linear logic about which I will talk is the multiplicative fragment plus mix, without constants. As we will see later on, the rule mix is necessary and fits naturally in our scheme, and its use is consistent with the results in [20], where it is employed.

5.1 Definition The system of *multiplicative linear logic without constants and with the rule mix* is denoted by MLL+mix and is such that:

1 *Formulae*, denoted by A, B and C are built over atoms according to

$$A ::= a \mid A \wp A \mid A \otimes A \mid \bar{A} \quad ,$$

where the binary connectives $\wp$ and $\otimes$ are called respectively *par* and *times* and $\bar{A}$ is the *negation* of A . When necessary, brackets are used to disambiguate expressions. Negation is defined by De Morgan laws:

$$\begin{aligned}
\overline{A \wp B} &= \bar{A} \otimes \bar{B} \quad , \\
\overline{A \otimes B} &= \bar{A} \wp \bar{B} \quad ,
\end{aligned}$$

and formulae are considered equivalent modulo the relation $=$.

2 *Sequents*, denoted by Σ , are expressions of the kind

$$\vdash A_1, \dots, A_h \quad ,$$

where $h \geq 0$ and the comma between the formulae $A_1, \dots, A_h$ stands for multiset union. Multisets of formulae are denoted by Φ and Ψ . For example, $\vdash \Phi, \Psi, A$ is a sequent whose formulae are those in $\Phi \uplus \Psi \uplus \{A\}_+$.

3 *Derivations*, denoted by Δ , are trees represented like

$$\begin{array}{c}
\Sigma_1 \dots \Sigma_h \\
\triangle \\
\Delta \\
\Sigma
\end{array} \quad ,$$

where $h \geq 0$, the sequents $\Sigma_1, \dots, \Sigma_h$ are called *premises*, Σ is the *conclusion*, and a finite number (possibly zero) of instances of the inference rules in Figure 12 are applied. A derivation with no premises is a *proof*, denoted by Π .

Linear logic formulae correspond to flat structures (i.e., structures where seq contexts do not appear) different from the unit.

5.2 Definition The function $\underline{\cdot}_\vee$ transforms the formulae of MLL+mix into flat structures according to the following inductive definition:

$$\begin{aligned}\underline{a}_\vee &= a \quad , \\ \underline{A \wp B}_\vee &= [\underline{A}_\vee, \underline{B}_\vee] \quad , \\ \underline{A \otimes B}_\vee &= (\underline{A}_\vee, \underline{B}_\vee) \quad .\end{aligned}$$

The domain of $\underline{\cdot}_\vee$ is extended to sequents this way:

$$\underline{\vdash A_1, \dots, A_{h_\vee}} = [\underline{A_1}_\vee, \dots, \underline{A_{h_\vee}}_\vee] \quad ,$$

where $h > 0$; let $\underline{\vdash}_\vee = \circ$.

The first, fairly obvious result of this section is about the ability of BV to mimic the derivations in MLL+mix. System FBV is, of course, enough. We need firstly to establish that if we restrict ourselves to using interaction rules with flat structures, then seq and coseq rules are no more necessary.

5.3 Lemma *If the rules $\text{id} \downarrow \frac{S\{\circ\}}{S[R, \bar{R}]}$ and $\text{id} \uparrow \frac{S(R, \bar{R})}{S\{\circ\}}$ are restricted to the case in which R is flat, then they are derivable for $\{\text{ai} \downarrow, \text{s}\}$ and $\{\text{ai} \uparrow, \text{s}\}$, respectively.*

Proof The proof is just a special case of the ones of Theorems 3.4.2 and 3.4.3. $\square$

5.4 Theorem *For every derivation $\frac{\Sigma_1 \dots \Sigma_h}{\Sigma}$ in MLL+mix there exists $\frac{(\underline{\Sigma_1}_\vee, \dots, \underline{\Sigma_h}_\vee)}{\underline{\Sigma}_\vee} \parallel_{\text{FBV}}$.*

Proof Let Δ be a given derivation in MLL+mix: let us proceed by structural induction on Δ . The following cases are possible:

- 1 If Δ is Σ , take $\underline{\Sigma}_\vee$.
- 2 If Δ is $\text{id} \frac{\vdash A, \bar{A}}{\vdash A, \bar{A}}$, for some formula A , take the proof in FBV obtained from $\text{id} \downarrow \frac{\circ \downarrow \frac{\vdash A, \bar{A}}{\vdash A, \bar{A}}}{[\underline{A}_\vee, \underline{\bar{A}}_\vee]}$ by unfolding the instance of $\text{id} \downarrow$, by Lemma 5.3.

- 3 If Δ has shape $\wp \frac{\frac{\Sigma_1 \dots \Sigma_h}{\vdash A, B, \Phi}}{\vdash A \wp B, \Phi}$, take $\frac{(\underline{\Sigma_1}_\vee, \dots, \underline{\Sigma_h}_\vee)}{\vdash A, B, \Phi_\vee} \parallel_{\text{FBV}}$, which exists by induction hypothesis.
- 4 Suppose that Δ has shape

$$\begin{array}{c} \frac{\frac{\Sigma'_1 \dots \Sigma'_k}{\vdash A, \Phi'} \quad \frac{\Sigma''_1 \dots \Sigma''_l}{\vdash B, \Psi'}}{\vdash A \otimes B, \Phi', \Psi'} \otimes \quad \text{or} \quad \text{mix} \frac{\frac{\Sigma'_1 \dots \Sigma'_k}{\vdash \Phi} \quad \frac{\Sigma''_1 \dots \Sigma''_l}{\vdash \Psi}}{\vdash \Phi, \Psi} \end{array} ,$$

where $k, l \geq 0$ and, for the derivation at the left, we can suppose that $\vdash \Phi = \vdash A, \Phi'$ and $\vdash \Psi = \vdash B, \Psi'$. Let us also suppose that $\Phi \neq \otimes_+ \neq \Psi$, otherwise the instance of mix would

be trivial. In both cases there are, by induction hypothesis, two derivations

$$\begin{array}{c} (\underline{\Sigma}'_{1\vee}, \dots, \underline{\Sigma}'_{k\vee}) \\ \Delta' \parallel_{\text{FBV}} \\ \vdash \underline{\Phi}_{\vee} \end{array} \quad \text{and} \quad \begin{array}{c} (\underline{\Sigma}''_{1\vee}, \dots, \underline{\Sigma}''_{l\vee}) \\ \Delta'' \parallel_{\text{FBV}} \\ \vdash \underline{\Psi}_{\vee} \end{array} .$$

Let $\Delta_1 = (\Delta', \underline{\Sigma}''_{1\vee}, \dots, \underline{\Sigma}''_{l\vee})$, where with (Δ', S) we denote the derivation obtained from Δ' by immersing every structure appearing in it in the hole of $(_, S)$; analogously, let $\Delta_2 = (\vdash \underline{\Phi}_{\vee}, \Delta'')$. We can take, respectively, the derivations

$$\begin{array}{c} (\underline{\Sigma}'_{1\vee}, \dots, \underline{\Sigma}'_{k\vee}, \underline{\Sigma}''_{1\vee}, \dots, \underline{\Sigma}''_{l\vee}) \\ \Delta_1 \parallel_{\text{FBV}} \\ (\vdash A, \underline{\Phi}'_{\vee}, \underline{\Sigma}''_{1\vee}, \dots, \underline{\Sigma}''_{l\vee}) \\ \Delta_2 \parallel_{\text{FBV}} \\ \text{s} \frac{(\vdash A, \underline{\Phi}'_{\vee}, \vdash B, \underline{\Psi}'_{\vee})}{[(\vdash A, \underline{\Phi}'_{\vee}, B_{\vee}), \vdash \underline{\Psi}'_{\vee}]} \\ \text{s} \frac{[(\underline{A}_{\vee}, B_{\vee}), \vdash \underline{\Phi}'_{\vee}, \vdash \underline{\Psi}'_{\vee}]}{[(\underline{A}_{\vee}, B_{\vee}), \vdash \underline{\Phi}'_{\vee}, \vdash \underline{\Psi}'_{\vee}]} \end{array} \quad \text{or} \quad \begin{array}{c} (\underline{\Sigma}'_{1\vee}, \dots, \underline{\Sigma}'_{k\vee}, \underline{\Sigma}''_{1\vee}, \dots, \underline{\Sigma}''_{l\vee}) \\ \Delta_1 \parallel_{\text{FBV}} \\ (\vdash \underline{\Phi}_{\vee}, \underline{\Sigma}''_{1\vee}, \dots, \underline{\Sigma}''_{l\vee}) \\ \Delta_2 \parallel_{\text{FBV}} \\ \text{s} \frac{(\vdash \underline{\Phi}_{\vee}, \vdash \underline{\Psi}_{\vee})}{[\vdash \underline{\Phi}_{\vee}, \vdash \underline{\Psi}_{\vee}]} \end{array} .$$

□

5.5 Corollary *If $\vdash A$ is provable in MLL+mix then $\underline{A}_{\vee}$ is provable in FBV.*

System FBV produces more, inherently different derivations from those of MLL+mix, as we saw in the introduction, because it is not bound to the application of rules to a main connective. A natural question at this point is whether it *proves more* than what MLL+mix does, and the answer is *no*. We need the inverse transformation of $\underline{\cdot}_{\vee}$.

5.6 Definition The function $\underline{\cdot}_{\perp}$ transforms non-unit normal flat structures into the formulae of MLL+mix according to the following inductive definition:

$$\begin{aligned} \underline{a}_{\perp} &= a \quad , \\ [\underline{R}, \underline{T}]_{\perp} &= \underline{R}_{\perp} \wp \underline{T}_{\perp} \quad , \\ (\underline{R}, \underline{T})_{\perp} &= \underline{R}_{\perp} \otimes \underline{T}_{\perp} \quad , \end{aligned}$$

where the formulae in the range of $\underline{\cdot}_{\perp}$ are to be considered modulo the equivalence relation generated by commutativity and associativity of $\wp$ and $\otimes$ (so that equivalent structures in the domain of $\underline{\cdot}_{\perp}$ are brought to equivalent formulae). This definition is extended to non-normal, non-unit flat structures in the obvious way.

5.7 Theorem *If a non-unit flat structure P is provable in FBV then $\vdash \underline{P}_{\perp}$ is provable in MLL+mix.*

Proof In the proof we will use the following known properties of provability in MLL+mix:

- 1 If $\vdash \bar{B}, C$ is provable then $\vdash \overline{A\{B\}}, A\{C\}$ is provable (where $A\{B\}$ is a context formula $A\{ \}$ whose hole has been filled with B).
- 2 The rule cut $\frac{\vdash A, \underline{\Phi} \quad \vdash \bar{A}, \underline{\Psi}}{\vdash \underline{\Phi}, \underline{\Psi}}$ is admissible for MLL+mix.

The first property is easily provable by induction on the structure of $A\{\circ\}$, the second by a standard first order cut elimination argument.

To prove the theorem let us proceed by induction on the length of the proof Π of P . In the base case $P = [a, \bar{a}]$:

$$\text{when } \Pi = \text{ai}\downarrow \frac{\circ \downarrow \frac{}{[a, \bar{a}]}}{\text{take } \frac{\text{id} \frac{}{\vdash a, \bar{a}}}{\wp \vdash a \wp \bar{a}}} .$$

There are two inductive cases, corresponding to the bottom rule instance in Π ; in both of them Π'' is the proof in MLL+mix corresponding, by induction hypothesis, to a proof Π' in FBV:

3 $P = S[a, \bar{a}]$, where $S\{\circ\} \neq \circ$, and the bottom inference rule instance is $\text{ai}\downarrow$: Let us firstly prove the following fact about provability in MLL+mix (very similar to the property 1 above): $\vdash \frac{S\{\circ\}}{S[a, \bar{a}]}$ is provable. Let us proceed by induction on the structure of $S\{\circ\}$. The inductive cases are

$$\frac{\text{id} \frac{}{\vdash \underline{T}, \underline{T}} \quad \frac{}{\vdash \underline{U}\{\circ\}, \underline{U}[a, \bar{a}]}}{\otimes \frac{}{\vdash \underline{T} \otimes \underline{U}\{\circ\}, \underline{T}, \underline{U}[a, \bar{a}]}} \quad \wp \frac{}{\vdash \underline{T} \otimes \underline{U}\{\circ\}, \underline{T} \wp \underline{U}[a, \bar{a}]}}{\wp \vdash \underline{T} \otimes \underline{U}\{\circ\}, \underline{T} \wp \underline{U}[a, \bar{a}]}} \quad \text{and} \quad \frac{\text{id} \frac{}{\vdash \underline{T}, \underline{T}} \quad \frac{}{\vdash \underline{U}\{\circ\}, \underline{U}[a, \bar{a}]}}{\otimes \frac{}{\vdash \underline{T}, \underline{U}\{\circ\}, \underline{T} \otimes \underline{U}[a, \bar{a}]}} \quad \wp \frac{}{\vdash \underline{T} \wp \underline{U}\{\circ\}, \underline{T} \otimes \underline{U}[a, \bar{a}]}}{\wp \vdash \underline{T} \wp \underline{U}\{\circ\}, \underline{T} \otimes \underline{U}[a, \bar{a}]}} ,$$

where $U\{\circ\} \neq \circ$. The base cases are

$$\frac{\text{id} \frac{}{\vdash \underline{T}, \underline{T}} \quad \frac{\text{id} \frac{}{\vdash a, \bar{a}}}{\wp \vdash a \wp \bar{a}}}{\text{mix} \frac{}{\vdash \underline{T}, \underline{T}, a \wp \bar{a}}} \quad \wp \frac{}{\vdash \underline{T}, \underline{T} \wp [a, \bar{a}]}}{\wp \vdash \underline{T}, \underline{T} \wp [a, \bar{a}]}} \quad \text{and} \quad \frac{\text{id} \frac{}{\vdash \underline{T}, \underline{T}} \quad \frac{\text{id} \frac{}{\vdash a, \bar{a}}}{\wp \vdash a \wp \bar{a}}}{\otimes \frac{}{\vdash \underline{T}, \underline{T} \otimes [a, \bar{a}]}} .$$

Coming back to our main proof:

$$\text{when } \Pi = \text{ai}\downarrow \frac{\Pi' \parallel_{\text{FBV}} \frac{S\{\circ\}}{S[a, \bar{a}]}}{\text{take } \frac{\text{cut} \frac{\frac{}{\vdash S\{\circ\}} \quad \frac{}{\vdash S\{\circ\}, S[a, \bar{a}]}}{\vdash S[a, \bar{a}]}} .$$

4 $P = S[(R, R'), T]$ and the bottom inference rule instance is s:

$$\text{when } \Pi = \text{s} \frac{\Pi' \parallel_{\text{FBV}} \frac{S([R, T], R')}{S[(R, R'), T]}}{\text{take } \frac{\text{cut} \frac{\frac{}{\vdash S([R, T], R')}} \quad \frac{}{\vdash S([R, T], R'), S[(R, R'), T]}}{\vdash S[(R, R'), T]}} ,$$

where we applied the property 1 above with the proof

$$\frac{\text{id} \frac{}{\vdash \bar{R}, R} \quad \text{id} \frac{}{\vdash \bar{R}', R'}}{\otimes \frac{}{\vdash \bar{R}, \bar{R}', R \otimes R'}} \quad \text{id} \frac{}{\vdash \bar{T}, T} \quad \otimes \frac{}{\vdash \bar{R} \otimes \bar{T}, \bar{R}', R \otimes R', T} \quad \wp \frac{}{\vdash \bar{R} \otimes \bar{T}, \bar{R}', (R \otimes R') \wp T} \quad \wp \frac{}{\vdash (\bar{R} \otimes \bar{T}) \wp \bar{R}', (R \otimes R') \wp T} .$$

Since cuts can be eliminated, by the property 2, the two proofs obtained in the inductive cases can be transformed into cut-free proofs of MLL+mix. $\square$

The correspondence between BV and MLL+mix, with respect to provability, is then complete on their common language:

5.8 Theorem *If a non-unit flat structure S is provable in BV then $\vdash \underline{S}_\perp$ is provable in MLL+mix.*

Proof It follows from Theorems 4.2.9 and 5.7. $\square$

5.9 Remark The correspondence of FBV to linear logic can be extended to multiplicative constants as follows. Let $\perp$ and 1 be, respectively, the units for the connectives $\wp$ and $\otimes$. Consider adding to MLL+mix the following three inference rules, where the first two are the usual rules for $\perp$ and 1 and the third, called mix0, as Abramsky and Jagadeesan do in [1], is a nullary version of mix:

$$\perp \frac{\vdash \Phi}{\vdash \perp, \Phi} \quad , \quad 1 \frac{}{\vdash 1} \quad , \quad \text{mix0} \frac{}{\vdash} \quad .$$

These rules make $\perp$ and 1 collapse: both $\vdash \perp, \perp$ and $\vdash 1, 1$ are provable. This means that we are left, essentially, with a single unit for both $\wp$ and $\otimes$, and we can map this unit into $\circ$ with no effort.

6 Conclusions

The question answered by this paper is: Is it possible to design a formal system in the tradition of proof theory, only based on simple properties that are universally recognised as fundamental ones in the concurrent management of information? This system should only come out of the analysis of some local properties of computation, like sending or receiving an atomic message, or preventing information to be exchanged by turning a finite, bounded switch. Not to float in a purely philosophical realm, I decided to apply these principles to the concrete problem of extending the multiplicative core of linear logic with a non-commutative operator. This investigation took the form of an experiment, of which I will give an account that also summarises the results achieved.

I departed from the rigid scheme of the sequent calculus and conceived what I call the ‘calculus of structures’. This is the tool I used for designing various formal systems, and it has the following differences from the sequent calculus:

- 1 Structures take the place of formulae and sequents, and the difference is only in the way we look at them. While formulae are trees built by binary connectives, structures are collections of atom occurrences where each couple of occurrences is bound by one and only one structural relation. Certain other laws hold for structures, whose purpose is to establish some good modularity properties. I have characterised the class of structures in terms of properties that structural relations must possess, and this move frees us from the idiosyncrasies of syntax: we can say that structures are *essentially* determined by their inner structural relations.

- 2 Since we have no more the idea of *main connective*, it is natural to think of inference rules that can act (i.e., rewrite) anywhere in structures. We can also limit ourselves to employing single-premiss rules only. This move only makes sense if we can show that we get an analogous feature to the *subformula property*. This in fact happens with our derivations, which can be read, bottom-up-wise, as progressively limiting the range of choices for interaction to happen. Vice versa, they can be read, top-down-wise, as making weaker and weaker assertions about times and places where interactions occur.

Then I used the calculus of structures for building a commutative/non-commutative system, in a very implicit way. The idea was: let us define the structural relations in which we are interested, and let us think of simple conservation laws that could make sense, and then let us try to find rules, whose complexity is minimal and which can implement our conservation laws.

The starting point was using system WMV (mentioned in Definition 3.3.2) on structures built over the three structural relations discussed in the introduction. System WMV is only based on the idea that two structures can freely merge provided that in doing so the original structural relations are conserved. I cannot think of a derivability notion simpler than this, which possesses our analogues of the subformula property. After having unfolded the two symmetric rules that I had and having added rules for interaction, I discovered that the cut rule was not admissible.

Interaction rules were untouchable, since what they do is simply to recognise that two structures are dual through negation, and this is primitive for me. Then I had to modify how I dealt with order, i.e., with structural relations, having in mind the only, limited objective of getting cut elimination. For some reason, solving this problem has been very difficult. The result is Definition 3.1.3, which adds to the previous merging law an axiom, which I call *integrity conservation*. Besides giving us cut elimination, this definition has the surprising property of producing exactly multiplicative linear logic (with mix) plus the desired non-commutative extension. This is remarkable because we get the core part of linear logic out of a very simple-minded, natural, purely syntactic and finitistic approach, essentially based on ideas of symmetry and conservation. Moreover, the resulting system BV is itself simple: order is managed by two rules, interaction by one, and it has an intuitive semantics which I do not hesitate to call *beautiful*.

The success of the calculus of structures in dealing with non-commutativity stems from a simple fact. Consider the seq rule $q\downarrow \frac{S\langle [R, T]; [R', T'] \rangle}{S[\langle R; R' \rangle, \langle T; T' \rangle]}$. In the sequent calculus

we are only able to mimic $\frac{S([R, T], [R', T'])}{S[\langle R; R' \rangle, \langle T; T' \rangle]}$, i.e., we get a derivation where the two substructures in the premiss are in a copar instead of a seq. This is too strong, because it prevents certain communications, between $S\{\circ\}$ and $[R, T]$ and between $S\{\circ\}$ and $[R', T']$, that are allowed in $q\downarrow$. The problem with the sequent calculus is that double-premiss rules always perform a conjunction between premisses, and we need something finer than that.

This does not mean that it is impossible to devise sequent systems that deal with non-commutativity together with commutativity. The recent works [2, 22], by Abrusci and

Ruet, are in fact a successful attempt at this task, and the first, as far as I know. The authors manage to build a multiplicative linear sequent system of four connectives, two commutative and two non-commutative, where each couple contains both a conjunction and a disjunction. Their system can be expanded to contain all of linear logic, therefore all of classical logic. The price they pay, in my opinion, is simplicity. Despite recent simplifications, their sequents are provided with a technically sophisticated but quite complex order structure. This conflicts with my belief that systems should be *simple* to be useful, but of course this is a very subjective matter.

Polakow and Pfenning, in [17], devise a natural deduction and an intuitionistic sequent system for a form of non-commutative linear logic, obtained as a refinement of intuitionistic linear logic. Their approach is essentially aimed at logic programming, with no attempt at an intimate coexistence of commutative and non-commutative relations, and their logic captures non-commutativity in hypotheses. The intrinsic, intuitionistic asymmetry of their systems prevents having an involutive negation. Yet, having a perfect symmetry, through negation, between two dual entities that communicate, is an absolute prerequisite, from the point of view of the logical foundations of concurrency. Then, I would say that the systems of Polakow and Pfenning are not a satisfying solution to our specific, foundational problem.

These two papers are the only that I know related to this work, apart from Retoré's ones, discussed in several places elsewhere in this paper. I do not know how to compare my system to those of others, if not on the basis of their promise of being groundwork for future applications and research. As a matter of fact, besides the formal systems that I discussed, there is the issue of having introduced a new proof-theoretical *methodology*.

Since the cut rule is admissible, our system is in principle a good candidate for such applications like automated deduction, logic programming, etc. The most direct comparison should be done with linear logic, of course, and then with the study of its uniform proofs. In this respect, the reference paper is Miller's [13], where it is defined the formal system *Forum*, which is equivalent to linear logic.

The biggest problem in the proof-construction field is eliminating as much as possible unnecessary, spurious non-determinism in the bottom-up search for a proof. *Forum* succeeds in eliminating all the non-determinism due to permutability of inference rules in linear logic, and employs a clever technique, called *focusing* and due to Andreoli [3], to keep the search for a proof nailed to certain choices done during a computation. Unfortunately, being intimately tied to the sequent calculus, *Forum* suffers from the very same context partitioning problem of the times rule. Therefore, it will be interesting for me to see whether my system can be subjected to the same treatment that Andreoli and Miller performed on linear logic, since this could lead to a system with all the good properties of *Forum*, with first-class non-commutativity, and without the context partitioning problem.

To this purpose, the situation in BV and its extensions is quite different from the one in the sequent calculus. As one of the examples in the introduction shows, reductions may occur everywhere in the structure, while building a derivation. There is not just a single active area corresponding to the main connective, or a few active areas corresponding to each of the formulae in a sequent. In fact, this is a big difference from the sequent calculus, and a definite improvement from the point of view of concurrency. On the other

hand, non-determinism can be undesirable, and the calculus of structures is much more non-deterministic than the sequent calculus.

Non-determinism in the sequent calculus is tamed by an analysis of rule permutations and their recombination. It's true that the calculus of structures yields more permutations (non-determinism), but at the same time it also offers more possibilities of recombination. This is testified by what we call *decomposition theorems*: derivations can be rearranged such that they are composed of a fixed sequence of segments, each of which is carried on by one of several disjoint subsystems of the original system. For example, multiplicative exponential linear logic exhibits seven disjoint subsystems which enjoy this property in several different rearrangements [10, 24]. This flexibility has no counterpart in other formalisms, like the sequent calculus, natural deduction or proof nets. A first success toward the taming of nondeterminism has been obtained by Bruscoli in [7], where she is able faithfully to capture a fragment of CCS [14] containing the prefix operator. So, I would say that it is far from obvious whether the calculus of structures is weak (or strong) at applications in proof construction.

The question is: Is there a *logical* reason to confine rewriting to certain, selected places of logical structures, just because they happen to be external? In the calculus of structures, all atoms in suitable structural relations may participate to rewriting at any time. This, in principle, could be important also for applications.

The conservation principles outlined in the paper are of course semantic ideas. Their application led me to the discovery of inference rules. It should be possible to work on these semantic ideas and to get a characterisation of derivability in larger and larger fragments. A problem is stated (3.3.4) whose solution would lead to a semantic characterisation of derivability in the combinatorial core of our system. The interest of the characterisation envisaged is in the fact that it would only involve checking a natural ordering relation induced by structural relations. I do not know if this investigation could give us also an *inspiring* semantics, even if the starting point seems good.

Games semantics is definitely something to explore. The paper [1] gives semantics to MLL+mix (multiplicative linear logic plus mix), which is a fragment of linear logic strictly included in my system. The semantics given there is fully complete for proof nets: every winning strategy denotes a unique cut-free proof net. Since derivations in the calculus of structures would interleave parts of their corresponding, hypothetical proof nets, as happens in the sequent calculus, it seems obvious firstly to establish proof nets for our system. In fact, one proof net usually stands for many derivations differing in the sequentialisation of the information in the net; therefore, proof nets are closer objects to semantics than derivations. The natural candidate for our case is Retoré's notion of proof net in [21], which, again, includes MLL+mix and features a non-commutative connective. That connective appears to be very similar to mine, and I hope the third projected paper on system BV will finally resolve the question.

Is it justified to introduce a new calculus, together with a new proof-theoretical methodology, simply to solve the problem of non-commutativity? The calculus of structures does more than just solving this problem. So far, the people in my group and myself have dealt with classical logic [6, 5], linear logic [9, 24, 23] and a conservative extension of both multiplicative exponential linear logic and system BV [10]. We are now able to

present classical and linear logic in an entirely local way, meaning that each application of a rule has a bounded computational cost, including contraction and the managing of additive contexts. More in general, we obtain properties of atomicity, locality and modularity, like the aforementioned decomposition theorems, that are not possible in any other proof-theoretical formalism. Thanks to atomicity of cut, cut elimination for classical logic can be proved with a stunningly concise and elegant proof [5].

I would like to express the following belief, if the reader forgives me some mysticism. I think that, in the construction of a formal system for computer science, *local* notions of semantics like those that I employed, for example simple laws of conservation, are more important than *global* ones, like a denotational semantics for derivations, say. The global perspective is in the end always enforced by cut elimination. In other words, I believe that we should start from the bottom, from the languages and the properties we want to model, and only later try to make a sense of it all. We can do so because cut elimination would anyway prevent us from making the worst mistakes. By doing this way I have been rewarded by the merge rules and their ‘space-temporal’ interpretation: it has been an outcome of unexpected beauty.

Acknowledgments

This work has been supported in part by a Marie Curie TMR grant that I used for a post-doc at INRIA, Nancy, in 1997/8. I started this project in 1994, encouraged by Dale Miller, who taught me a lot about proof theory. Horst Reichel and Enno Sandner listened to me and put interesting questions about preliminary versions of this paper. Giorgio Levi has been very encouraging throughout this endeavour. This work would not be so minimalist in spirit had I not listened to and enjoyed so much the music of Michael Nyman and Philip Glass. Jim Lipton carefully read the introduction and discussed with me several related issues; he has been very helpful. Pietro Di Gianantonio has been a great source of encouragement, especially because he liked this paper; he directly inspired the splitting theorem. Alwen Tiu, Janis Voigtländer, Kai Brännler, Lutz Straßburger, Paola Bruscoli and Steffen Hölldobler checked many details; Alwen, Kai and Lutz also found several mistakes in early versions of the cut elimination argument, and this eventually led me to discover splitting. I warmly thank Christian Retoré for having introduced me to the fascinating problem of non-commutativity in linear logic; he deeply influenced my ideas on this subject. Finally, I thank the two referees for the unbelievable amount of work they dedicated to my paper; its present shape owes much to them.

References

- [1] Samson Abramsky and Radha Jagadeesan. Games and full completeness for multiplicative linear logic. *Journal of Symbolic Logic*, 59(2):543–574, Jun 1994.
- [2] V. Michele Abrusci and Paul Ruet. Non-commutative logic I: The multiplicative fragment. *Annals of Pure and Applied Logic*, 101(1):29–64, 2000.

- [3] Jean-Marc Andreoli. Logic programming with focusing proofs in linear logic. *Journal of Logic and Computation*, 2(3):297–347, 1992.
- [4] Denis Bechet, Philippe De Groote, and Christian Retoré. A complete axiomatisation of the inclusion of series-parallel orders. In H. Comon, editor, *Rewriting Techniques and Applications, RTA '97*, volume 1232 of *Lecture Notes in Computer Science*, pages 230–240. Springer-Verlag, 1997.
- [5] Kai Brännler. Atomic cut elimination for classical logic. In M. Baaz and J. A. Makowsky, editors, *CSL 2003*, volume 2803 of *Lecture Notes in Computer Science*, pages 86–97. Springer-Verlag, 2003. URL: <http://www.iam.unibe.ch/~kai/Papers/AtomicCutElimination-short.pdf>.
- [6] Kai Brännler and Alwen Fernanto Tiu. A local system for classical logic. In R. Nieuwenhuis and A. Voronkov, editors, *LPAR 2001*, volume 2250 of *Lecture Notes in Artificial Intelligence*, pages 347–361. Springer-Verlag, 2001. URL: <http://www.iam.unibe.ch/~kai/Papers/LocalClassicalLogic-lpar.pdf>.
- [7] Paola Bruscoli. A purely logical account of sequentiality in proof search. In Peter J. Stuckey, editor, *Logic Programming, 18th International Conference*, volume 2401 of *Lecture Notes in Computer Science*, pages 302–316. Springer-Verlag, 2002. URL: <http://www.ki.inf.tu-dresden.de/~paola/bvl/bvl.pdf>.
- [8] Jean-Yves Girard. Linear logic. *Theoretical Computer Science*, 50:1–102, 1987.
- [9] Alessio Guglielmi and Lutz Straßburger. Non-commutativity and MELL in the calculus of structures. In L. Fribourg, editor, *CSL 2001*, volume 2142 of *Lecture Notes in Computer Science*, pages 54–68. Springer-Verlag, 2001. URL: <http://www.ki.inf.tu-dresden.de/~guglielm/res/pap/NoncMELLCoS.pdf>.
- [10] Alessio Guglielmi and Lutz Straßburger. A non-commutative extension of MELL. In M. Baaz and A. Voronkov, editors, *LPAR 2002*, volume 2514 of *Lecture Notes in Artificial Intelligence*, pages 231–246. Springer-Verlag, 2002. URL: <http://www.ki.inf.tu-dresden.de/~lutz/NEL.pdf>.
- [11] Joshua S. Hodas and Dale Miller. Logic programming in a fragment of intuitionistic linear logic. *Information and Computation*, 110(2):327–365, May 1994.
- [12] Joachim Lambek. The mathematics of sentence structure. *American Mathematical Monthly*, 65:154–169, 1958.
- [13] Dale Miller. Forum: A multiple-conclusion specification logic. *Theoretical Computer Science*, 165:201–232, 1996.
- [14] Robin Milner. *Communication and Concurrency*. International Series in Computer Science. Prentice Hall, 1989.
- [15] Rolf H. Möhring. Computationally tractable classes of ordered sets. In I. Rival, editor, *Algorithms and Order*, pages 105–194. Kluwer Academic Publishers, Dordrecht, 1989.
- [16] Nuel D. Belnap, Jr. Display logic. *Journal of Philosophical Logic*, 11:375–417, 1982.
- [17] Jeff Polakow and Frank Pfenning. Relating natural deduction and sequent calculus for intuitionistic non-commutative linear logic. *Electronic Notes in Theoretical Computer Science*, 20, 1999. URL: <http://www.elsevier.nl/locate/entcs/volume20.html>.
- [18] Greg Restall. *An Introduction to Substructural Logics*. Routledge, 2000.
- [19] Christian Retoré. *Réseaux et Séquents Ordonnés*. Thèse de Doctorat, spécialité mathématiques, Université Paris 7, February 1993.
- [20] Christian Retoré. Pomset logic: A non-commutative extension of classical linear logic. In Ph. de Groote and J. R. Hindley, editors, *TLCA '97*, volume 1210 of *Lecture Notes in Computer Science*, pages 300–318, 1997.
- [21] Christian Retoré. Pomset logic as a calculus of directed cographs. In V. M. Abrusci and C. Casadio, editors, *Dynamic Perspectives in Logic and Linguistics*, pages 221–247. Bulzoni, Roma, 1999. Also available as INRIA Rapport de Recherche RR-3714.

- [22] Paul Ruet. Non-commutative logic II: Sequent calculus and phase semantics. *Mathematical Structures in Computer Science*, 10:277–312, 2000.
- [23] Lutz Straßburger. A local system for linear logic. In M. Baaz and A. Voronkov, editors, *LPAR 2002*, volume 2514 of *Lecture Notes in Artificial Intelligence*, pages 388–402. Springer-Verlag, 2002. URL: <http://www.ki.inf.tu-dresden.de/~lutz/lrs-lpar.pdf>.
- [24] Lutz Straßburger. MELL in the calculus of structures. *Theoretical Computer Science*, 309:213–285, 2003. URL: <http://www.ki.inf.tu-dresden.de/~lutz/els.pdf>.
- [25] Alwen Fernanto Tiu. Properties of a logical system in the calculus of structures. Technical Report WV-01-06, Technische Universität Dresden, 2001. URL: <http://www.cse.psu.edu/~tiu/thesisc.pdf>.
- [26] A.S. Troelstra and H. Schwichtenberg. *Basic Proof Theory*, volume 43 of *Cambridge Tracts in Theoretical Computer Science*. Cambridge University Press, 1996.

Index of Symbols

- | | |
|---|---|
| <ul style="list-style-type: none"> $\circ$, unit, 7 $\langle S_1; \dots; S_h \rangle$, seq structure, 7 $[S_1, \dots, S_h]$, par structure, 7 $(S_1, \dots, S_h)$, copar structure, 7 $\bar{S}$, negation of structure S, 7 $S\{ \}$, structure context, 7 $=$, syntactic equivalence on structures, 8 $\triangleleft_S$, ‘seq’ structural relation of S, 10 $\triangleright_S$, ‘coseq’ structural relation of S, 10 $\downarrow_S$, ‘par’ structural relation of S, 10 $\uparrow_S$, ‘copar’ structural relation of S, 10 α, cardinality of set α, 14 $R \diamond T$, merge set of R and T, 21 $R \diamond T$, merge set of R and T (recursive definition), 22 $\leq$, relation comparing ‘energy’ of structures, 30 $\circ\downarrow$, ‘unit’ inference rule, 36 $\prec$, lexicographic order, 39 $\wp$, linear logic connective ‘par’, 52 $\wp$, linear logic inference rule ‘par’, 52 $\otimes$, linear logic connective ‘times’, 52 $\otimes$, linear logic inference rule ‘times’, 52 $\dot{\downarrow}_V$, transformation from linear logic formulae to structures, 53 $\emptyset_+$, empty multiset, 53 $\dot{\downarrow}_L$, transformation from structures to linear logic formulae, 54 $\perp$, linear logic disjunctive multiplicative unit, 56 1, linear logic conjunctive multiplicative unit, 56 | <ul style="list-style-type: none"> A, linear logic formula, 52 a, atom, 7 $\text{ai}\downarrow$, ‘atomic interaction’ inference rule, 31 $\text{ai}\uparrow$, ‘atomic cointeraction’ inference rule, 31 B, linear logic formula, 52 b, atom, 7 BV, basic system $\vee$, 37 C, linear logic formula, 52 c, atom, 7 cut, linear logic inference rule ‘cut’, 54 Δ, derivation, 18 Δ, linear logic derivation, 52 f, classical logic constant ‘false’, 9 FBV, flat BV, 51 Φ, multiset of linear logic formulae, 52 ϕ, set of structures, 22 Forum, Miller’s formal system for linear logic, 58 $\text{g}\downarrow$, ‘merge’ inference rule, 24 $\text{g}\uparrow$, ‘comerge’ inference rule, 24 $\text{i}\downarrow$, ‘interaction’ inference rule, 31 $\text{i}\uparrow$, ‘cointeraction’ inference rule, 31 id, linear logic inference rule ‘identity’, 52 LK, Gentzen’s sequent system of classical logic, 17 |
|---|---|

- mix, linear logic inference rule ‘mix’, 52
- mix0, linear logic inference rule ‘nullary mix’, 56
- MLL+mix, multiplicative linear logic plus mix, 52
- μ , set of atom occurrences, 11
- MV, merge system $\mathcal{V}$, 24
- ν , set of atom occurrences, 11
- occ S , set of atom occurrences in S , 10
- P , structure, 7
- Π , proof, 37
- Π , linear logic proof, 52
- π , inference rule name, 18
- Ψ , multiset of linear logic formulae, 52
- ψ , set of structures, 22
- Q , structure, 7
- $q\downarrow$, ‘seq’ inference rule, 26
- $q\uparrow$, ‘coseq’ inference rule, 26
- R , structure, 7
- ρ , inference rule name, 18
- S , structure, 7
- $\mathcal{S}$, formal system, 18
- s, ‘switch’ inference rule, 26
- SBV, symmetric basic system $\mathcal{V}$, 35
- SBVc, core symmetric basic system $\mathcal{V}$, 26
- Σ , linear logic sequent, 52
- σ , structural relation, 12
- T , structure, 7
- t, classical logic constant ‘true’, 9
- U , structure, 7
- V , structure, 7
- w S , relation web of S , 11
- wg $\downarrow$, ‘weak merge’ inference rule, 31
- wg $\uparrow$, ‘weak comerge’ inference rule, 31
- WMV, weak merge system $\mathcal{V}$, 31
- ws, ‘weak switch’ inference rule, 31
- X , structure, 7
- ξ , set of atom occurrences, 13
- ζ , web candidate, 13