Residuation in mixed Lambek systems*

Michael Moortgat
Research Institute for Language and Speech (OTS)
Trans 10, Utrecht
moortgat@let.ruu.nl

Abstract

In this paper we compare grammatical inference in the context of *simple* and of *mixed* Lambek systems. Simple Lambek systems are obtained by taking the logic of residuation for a family of multiplicative connectives $/, \bullet, \backslash$, together with a package of structural postulates characterizing the resource management properties of the $\bullet$ connective. Different choices for Associativity and Commutativity yield the familiar Lambek systems **NL**, **L**, **NLP**, **LP**. Semantically, a simple Lambek system is a unimodal logic: the connectives get a Kripke style interpretation in terms of a single ternary accessibility relation modeling the notion of linguistic composition for each individual system.

The simple systems each have their virtues in linguistic analysis. But none of them in isolation provides a basis for a full theory of grammar. In the second part of the paper, we consider two types of *mixed* Lambek systems. The first type is obtained by generalizing residuation to families of n-ary connectives, and by putting together the different arities in one logic. We focus on residuation for unary connectives, hence on mixed (2,3) frames, as these already represent the complexities in full. We prove a number of elementary logical results for unary families of residuated connectives and their combination with binary families.

The second type of mixed system is obtained by combining a number of unimodal systems into one multimodal logic. The combined multimodal logic is set up in such a way that the individual resource management properties of the constituting logics are preserved. But the inferential capacity of the mixed logic is greater than the sum of its component parts through the addition of frame conditions with the corresponding interaction postulates regulating the *communication* between the component logics.

The Appendix presents a number of new proof-theoretic invariants for the logics discussed.

---

*The materials in this paper prepare the ground for the sections on multiplicative connectives in a chapter on categorial grammar for the Handbook of Logic and Language (Van Benthem and ter Meulen (eds), Elsevier, to appear). Earlier versions of the paper were presented at the 'Deduction and Language' workshop, London, March 1994, and the Rome workshop on Lambek Calculus, May 1994. I thank the editors of the Handbook and the audiences at these workshops for comments. I have greatly benefited from discussion, and joint work, with Natasha Kurtonina, Glyn Morrill and Dick Oehrle. All errors are my own.

1 Residuation: simple Lambek systems

This paper traces the ramifications of Residuation in a variety of categorial type logics. To set the scene, we briefly present the concept of residuation in general algebraic terms as it arises in the study of order-preserving mappings. In Moortgat and Oehrle ([24]) the reader can find a more thorough treatment with reference to the source material (such as [12, 6]).

Residuated pairs

Let $\mathcal{A} = (A, \sqsubseteq_A)$ and $\mathcal{B} = (B, \sqsubseteq_B)$ be partially ordered sets. Consider a pair of functions $f : A \mapsto B$ and $g : B \mapsto A$. The pair (f, g) is called *residuated* if the inequalities of $(\star)$ hold. Alternatively, a pair of functions (f, g) is characterized as residuated by requiring f and g to be isotone $(\dagger)$, and by having the composition of the functions satisfy the inequalities of $(\ddagger)$.

$$(\star) \qquad fx \sqsubseteq_B y \quad \text{iff} \quad x \sqsubseteq_A gy$$

$$(\dagger) \quad \text{if } x \sqsubseteq_A y \ (x \sqsubseteq_B y) \text{ then } fx \sqsubseteq_B fy \ (gx \sqsubseteq_A gy)$$
$$(\ddagger) \qquad fgx \sqsubseteq_B x, \quad x \sqsubseteq_A gfx$$

Dunn's papers on 'gaggle theory' ([10, 11]) provide an excellent survey of the many guises under which Residuation presents itself in (intuitionistic, modal, relevance, dynamic, temporal, linear,...) logic, and in Lambek style type logics — the object of our investigation which we now turn to.

Binary multiplicatives

We start with a quick review of the landscape of binary multiplicative operators. This is extremely well-trodden ground, and the present section contains nothing new. But it sets the agenda for our exploration of more adventurous territory in §2.1 and §2.2

Consider the language $\mathcal{F}$ of category formulae (or: types) of a simple Lambek system. $\mathcal{F}$ is obtained by closing a set of prime formulae (or: basic types, e.g. *s, np, n, ...*) under binary connectives (or: type forming operators) $/, \bullet, \backslash$.

$$\mathcal{A} ::= p_0 \mid p_1 \mid p_2 \mid \ldots$$

$$\mathcal{F} ::= \mathcal{A} \mid \mathcal{F}/\mathcal{F} \mid \mathcal{F} \bullet \mathcal{F} \mid \mathcal{F}\backslash\mathcal{F}$$

Type formulae have a quite general interpretation in the power set algebra of Kripke style relational structures — ternary relational structures in the case of the binary connectives ([8]). A *ternary frame* is a structure $\langle W, R^3 \rangle$. W in the application we envisage here is to be thought of as a set of *linguistic resources* (or: signs, pieces of multidimensional linguistic information). The accessibility relation R can be understood as representing linguistic composition: $Rzxy$ holds in case one can fuse together the information of signs x and y into a sign z. We obtain a model by adding a valuation v sending prime formulae to subsets of W and satisfying the clauses below for compound formulae.

$$\begin{array}{lcl} v(A \bullet B) & = & \{z \mid \exists x \exists y [Rzxy \ \& \ x \in v(A) \ \& \ y \in v(B)]\} \\ v(C/B) & = & \{x \mid \forall y \forall z [(Rzxy \ \& \ y \in v(B)) \Rightarrow z \in v(C)]\} \\ v(A\backslash C) & = & \{y \mid \forall x \forall z [(Rzxy \ \& \ x \in v(A)) \Rightarrow z \in v(C)]\} \end{array}$$

Turning now to the *logic* of these type systems, we are interested in characterizing a relation of derivability between formulae such that $A \rightarrow B$ is provable iff $v(A) \subseteq$

$v(B)$. It is not difficult to check that given the above interpretation of compound formulae, the following laws determine the properties of $\bullet$ vis à vis $/, \backslash$ with respect to derivability.

$$\text{(RES)} \qquad A \to C/B \quad \Longleftrightarrow \quad A \bullet B \to C \quad \Longleftrightarrow \quad B \to A\backslash C$$

The pairs of connectives $(\bullet, /)$ and $(\bullet, \backslash)$ are easily recognized as the binary incarnations of the notion of residuation just defined for the case of unary operations f, g. Interpret the partially ordered set $\mathcal{A}$ $(= \mathcal{B})$ as the set of type formulae $\mathcal{F}$, ordered by derivability $\to$ (i.e. set-theoretic inclusion, semantically). For the *right residual* pair $(\bullet, /)$ we can read f as $- \bullet B$ and g as $-/B$, i.e. the product and division operations indexed by the fixed type B. The defining biconditional $fx \leq y$ iff $x \leq gy$ then becomes (†) below. Similarly for the *left residual* pair $(\bullet, \backslash)$, where we read f as $A \bullet -$ and g as $A\backslash-$, and obtain (‡).

$$\begin{array}{lrcl} (\dagger) & A \bullet B \to C & \Longleftrightarrow & A \to C/B \\ (\ddagger) & A \bullet B \to C & \Longleftrightarrow & B \to A\backslash C \end{array}$$

Putting things together, we see that the anatomy of the most elementary Lambek type logic is given by the basic properties of the derivability relation (Reflexivity, Transitivity) plus the Residuation Laws establishing the relation between $\bullet$ and the two implications $/, \backslash$. Below the axiomatic presentation of the system known as **NL**. Following [22], we add combinator proof terms: they will provide a compact way of referring to complete deductions later on. Via a canonical model construction Došen [8] obtains the elementary soundness and completeness result: in **NL** provability coincides with semantic inclusion for all ternary frames and all interpretations v.

NL: the pure logic of residuation

Combinator proof terms. We write $f : A \to B$ for a proof of the inclusion $v(A) \subseteq v(B)$.

$$\mathbf{id}_A : A \to A \qquad \frac{f : A \to B \quad g : B \to C}{g \circ f : A \to C}$$

$$\frac{f : A \bullet B \to C}{\beta(f) : A \to C/B} \qquad \frac{f : A \bullet B \to C}{\gamma(f) : B \to A\backslash C}$$

$$\frac{g : A \to C/B}{\beta^{-1}(g) : A \bullet B \to C} \qquad \frac{g : B \to A\backslash C}{\gamma^{-1}(g) : A \bullet B \to C}$$

Structural postulates, constraints on frames

Starting from the pure logic of residuation **NL** one can unfold a landscape of categorial type logics by gradually relaxing structure sensitivity in a number linguistically relevant dimensions. Below we consider the dimensions of linear precedence (order sensitivity) and immediate dominance (constituent sensitivity). Adding the structural postulates for Associativity or Commutativity (or both) to the pure logic of residuation, one obtains the systems **L**, **NLP**, **LP**.

$$\begin{array}{lc} \text{ASS} & A \bullet (B \bullet C) \longleftrightarrow (A \bullet B) \bullet C \\ \text{COMM} & A \bullet B \to B \bullet A \end{array}$$

Using Correspondence Theory [4] one computes frame conditions restricting the interpretation of R^3 for the different structural postulates. Došen's completeness result for **NL** is then extended to the stronger logics by restricting the attention to ASS (**L**), COMM (**NLP**) or ASS+COMM frames (**LP**).

ASS	$A \bullet (B \bullet C) \longleftrightarrow (A \bullet B) \bullet C$	$\exists t.Rtxy \ \&\ Rutz \Longleftrightarrow \exists v.Rvyz \ \&\ Ruxv$
COMM	$A \bullet B \rightarrow B \bullet A$	$Rzxy \Longleftrightarrow Rzyx$

Gentzen calculus

The axiomatic presentation is the proper vehicle for model-theoretic investigation of the logics we have considered: it closely follows the semantics, thus providing a suitable basis for 'easy' completeness results. Proof-theoretically the axiomatic presentation has a serious drawback: because it is essentially based on Transitivity, it does not offer an appropriate basis for proof search. For proof-theoretic investigation of the categorial type logics one introduces a Gentzen presentation, and proves a Cut Elimination result, with its corollaries of decidability and the subformula property. Of course, one has to establish the equivalence between the axiomatic and the Gentzen presentations of the logic for all this to make sense. For **L** Lambek [21] has established the essential results. They have been extended to the full landscape of type logics in [19, 9].

In the axiomatic presentation, we consider arrows $A \rightarrow B$ with $A, B \in \mathcal{F}$. In Gentzen presentation, the derivability relation is stated to hold between a *term* $\mathcal{T}$ (the antecedent) and a type formula (the succedent). A Gentzen term is a structured configuration of formulae — a structured database, in the terminology of Gabbay [13]. The term language is defined inductively as $\mathcal{T} ::= \mathcal{F} \mid (\mathcal{T}, \mathcal{T})$. The structural connective $(\cdot, \cdot)$ in the term language tells you how to put together structured databases Δ_1 and Δ_2 into a structured database (Δ_1, Δ_2). The structural connective mimics the logical connective $\bullet$ in the type language. A sequent is a pair (Γ, A) with $\Gamma \in \mathcal{T}$ and $A \in \mathcal{F}$, written as $\Gamma \Rightarrow A$.

To compare the two presentations, we define the formula equivalent $\Delta^\flat$ of a structured database Δ. Let $(\Delta_1, \Delta_2)^\flat = \Delta_1^\flat \bullet \Delta_2^\flat$, and $A^\flat = A$ for $A \in \mathcal{F}$. The Gentzen presentation can be show to be equivalent to the combinator axiomatisation in the sense of the following proposition from [21].

> Every combinator $f : A \rightarrow B$ gives a proof of $A \Rightarrow B$, and every proof of a sequent $\Gamma \Rightarrow B$ gives a combinator $f : \Gamma^\flat \rightarrow B$.

As was the case for the combinator presentation, the sequent architecture consists of three components: (i) [Ax] and [Cut] capture the basic properties of the derivability relation '$\Rightarrow$': reflexivity and contextualized transitivity for the 'surgical' Cut, (ii) each connective comes with two *logical rules*: a rule of use introducing the connective to the left of '$\Rightarrow$' and a rule of proof introducing it on the right of '$\Rightarrow$', finally (iii) there is a block of *structural rules*, possibly empty, with different packages of structural rules resulting in systems with different resource management properties.

Gentzen presentation: structured databases

Sequents $\mathcal{T} \Rightarrow \mathcal{F}$ where $\mathcal{T} ::= \mathcal{F} \mid (\mathcal{T}, \mathcal{T})$. Notation: $\Gamma[\Delta]$ for an antecedent term Γ containing a distinguished occurrence of the subterm Δ.

$$[\mathrm{Ax}]\frac{}{A \Rightarrow A} \qquad \frac{\Delta \Rightarrow A \quad \Gamma[A] \Rightarrow C}{\Gamma[\Delta] \Rightarrow C}[\mathrm{Cut}]$$

$$[/\mathrm{R}]\frac{(\Gamma, B) \Rightarrow A}{\Gamma \Rightarrow A/B} \qquad \frac{\Delta \Rightarrow B \qquad \Gamma[A] \Rightarrow C}{\Gamma[(A/B, \Delta)] \Rightarrow C}[/\mathrm{L}]$$

$$[\backslash\mathrm{R}]\frac{(B, \Gamma) \Rightarrow A}{\Gamma \Rightarrow B\backslash A} \qquad \frac{\Delta \Rightarrow B \qquad \Gamma[A] \Rightarrow C}{\Gamma[(\Delta, B\backslash A)] \Rightarrow C}[\backslash\mathrm{L}]$$

$$[\bullet\mathrm{L}]\frac{\Gamma[(A, B)] \Rightarrow C}{\Gamma[A \bullet B] \Rightarrow C} \qquad \frac{\Gamma \Rightarrow A \qquad \Delta \Rightarrow B}{(\Gamma, \Delta) \Rightarrow A \bullet B}[\bullet\mathrm{R}]$$

Resource sensitive structural rules

Permutation and Associativity.

$$\frac{\Gamma[(\Delta_2, \Delta_1)] \Rightarrow A}{\Gamma[(\Delta_1, \Delta_2)] \Rightarrow A}[\mathrm{P}] \qquad \frac{\Gamma[((\Delta_1, \Delta_2), \Delta_3)] \Rightarrow A}{\overline{\Gamma[(\Delta_1, (\Delta_2, \Delta_3))] \Rightarrow A}}[\mathrm{A}]$$

For the logics **L** and **LP** where $\bullet$ is associative, resp. associative and commutative, explicit application of the structural rules is generally compiled away by means of syntactic sugaring of the sequent language. Antecendent terms then take the form of sequences of formulae $\mathcal{T} ::= \mathcal{F}, \ldots, \mathcal{F}$ where the comma is now of variable arity, rather than a binary structural connective. Reading these antecedents as sequences, one avoids explicit reference to the structural rule of Associativity; reading them as multisets, one also makes Permutation implicit.

Characteristic theorems, derived rules of inference

We close this overview with an inventory of theorems and derived inference rules for the various logics.

1. Application: $A/B \bullet B \to A$, $B \bullet B\backslash A \to A$
2. Co-application: $A \to (A \bullet B)/B$, $A \to B\backslash(B \bullet A)$
3. Monotonicity $\bullet$: if $A \to B$ and $C \to D$, then $A \bullet C \to B \bullet D$
4. Isotonicity $\cdot/C$, $C\backslash\cdot$: if $A \to B$, then $A/C \to B/C$ and $C\backslash A \to C\backslash B$
5. Antitonicity $C/\cdot$, $\cdot\backslash C$: if $A \to B$, then $C/B \to C/A$ and $B\backslash C \to A\backslash C$
6. Lifting: $A \to B/(A\backslash B)$, $A \to (B/A)\backslash B$
7. Geach (main functor): $A/B \to (A/C)/(B/C)$, $B\backslash A \to (C\backslash B)\backslash(C\backslash A)$
8. Geach (secondary functor): $B/C \to (A/B)\backslash(A/C)$, $C\backslash B \to (C\backslash A)/(B\backslash A)$
9. Composition: $A/B \bullet B/C \to A/C$, $C\backslash B \bullet B\backslash A \to C\backslash A$
10. Restructuring: $(A\backslash B)/C \longleftrightarrow A\backslash(B/C)$
11. (De)Currying: $A/(B \bullet C) \longleftrightarrow (A/C)/B$, $(A \bullet B)\backslash C \longleftrightarrow B\backslash(A\backslash C)$
12. Permutation: if $A \to B\backslash C$ then $B \to A\backslash C$

13. Exchange: $A/B \longleftrightarrow B\backslash A$

14. Preposing/Postposing: $A \to B/(B/A)$, $A \to (A\backslash B)\backslash B$

15. Mixed Composition: $A/B \bullet C\backslash B \to C\backslash A$, $B/C \bullet B\backslash A \to A/C$

Items (1) to (5) are valid in the weakest logic **NL**. They provide an alternative to RES to express the fact that $(\bullet,/)$ and $(\bullet,\backslash)$ are residuated pairs, cf. (†,‡) of Def 1.1. Lifting is the closest one can get to (2) in 'product-free' type languages, i.e. type languages where the role of the product operator (generally left implicit) is restricted to glue together types on the left-hand side of the arrow. Items (7) to (11) mark the transition to **L**: their derivation involves the structural postulate of associativity for $\bullet$. Rule (12) is characteristic for systems with a commutative $\bullet$, **NLP** and **LP**. From (12) one immediately derives the collapse of the implications / and \, (13). As a result of this collapse, one gets variants of the earlier theorems obtained by substituting subtypes of the form A/B by $B\backslash A$ or vice versa. Examples are (14), an **NLP** variant of Lifting, or (15), an **LP** variant of Composition.

The simple Lambek systems each have their merits and their limitations when it comes to grammatical analysis. In Combinatory Categorial Grammar ([32]) instances of the full scala of type transitions above live together. In the presence of Residuation, such promiscuity has unpleasant consequences. The point here is that when one moves from a logic with a higher degree of structure sensitivity to a system with more relaxed resource management, one immediately looses sensitivity for the relevant structural parameter of the weaker logic. For example: the system **NL** has a hierarchically structured database which fully respects constituent structure. For cases of so-called non-constituent coordination, one would like to relax constituent structure. One could try to achieve this by adding Composition (or the Geach laws) to **NL**. But the addition of such postulates makes **NL** collapse into **L**: from Geach one easily obtains the unconditional Associativity postulate for $\bullet$ via Residuation. We leave this as an exercise for the reader.

In the following sections we try to develop a more articulate view on mixed styles of inference where such collapse is avoided.

2 Residuation in mixed logics

2.1 Mixed inference: combining 1-ary and 2-ary families

In the recent literature one finds a panoply of unary operators in addition to the multiplicatives we have discussed in the previous section:

- The 'domain modality' $\Box$ of Hepple [16]. Used to enforce locality constraints in terms of syntactic domains. Generalized to multimodal $\Box_i$ to distinguish domain barriers of different strength. A syntactiv version of the semantically interpreted intensionality operator of Morrill [26].

- The 'bracket operators' $[], []^{-1}$ of Morrill [27, 28]. Serving the same purpose, but implementing the idea in a quite different way.

- The $\Diamond$ operator of Morrill [27], declaring argument positions as licensing extraction.

- The Edinburgh 'structural modalities'. Modelled after Linear Logic's '!' (bang) operator. Providing controlled access to more linguistically relevant structural options such as Permutation, Associativity.

Although the various operators are introduced with intended semantics and (Gentzen, natural deduction) proof theory, discussion of the *relation* between the model theory and the proof theory is lacking. This can easily lead to confusion. For example, the domain modalities and the Edinburgh structural modalities have been introduced with an S4 proof theory, borrowed from LL, and a subalgebra semantics in a general groupoid setting. But for the subalgebra semantics, the S4 proof theory is inadequate ([34]): it imposes semantic constraints which the subalgebra interpretation cannot support. There is a choice here: if it is the proof theory one likes, one should provide an adequate semantics with a soundness and completeness result ([20]). Or if the structure of the semantics is the linguistically important factor, one should provide appropriate proof theory ([33]).

Our aim in this section is to provide a general framework within which the different proposals for unary operators can be compared. The key concept, again, is residuation. We extend the language of binary multiplicatives with a unary pair of residual operators $\diamond$, $\Box^{\downarrow}$ and establish some elementary logical results for the extended language. Parallel to our treatment of the binary multiplicatives $/, \bullet, \backslash$ in the previous section, we start from the pure logic of residuation for $\diamond$, $\Box^{\downarrow}$, and gradually add structural postulates in view of the linguistic applications.

Items (1) and (3), (4) closely follow Lambek's [21] treatment of the binary multiplicative operators. We refer to Kurtonina [20] for a thorough investigation of the further logical ramifications of the matters dealt with here.

1. Axiomatic ('combinator') presentations of the pure logic of residuation for $\diamond$, $\Box^{\downarrow}$.

2. Soundness and completeness via the Došen canonical model construction.

3. Gentzen presentation, equivalence between the axiomatic and the Gentzen presentation.

4. Cut elimination for the Gentzen presentation. Decidability, subformula property.

5. Structural postulates $T, 4, K$. Items (1)–(3) for the systems with a choice from $\mathcal{P}(\{T, 4, K\})$.

Residuation: n-ary generalisation

The concept of residuation can be readily generalized to the case of n-ary connectives, as is shown in [10] in the general logical setting. Discussion of such generalizations for categorial type logics can be found in [7] and [24]. In the context of Kripke style frame semantics, we have n-ary products interpreted via $n+1$-ary accessibility relations. These products have a residual implication for each of their n factors. Let us write $f_{\bullet}(A_1, \ldots, A_n)$ for the product and $f^{i}_{\rightarrow}(A_1, \ldots, A_n)$ for the i-th place residual. And let us define $R^{-i} y_i, y_1, \ldots, x, \ldots, y_n$ iff $Rx, y_1, \ldots, y_i, \ldots, y_n$ to facilitate the statement of the interpretation clauses. Valuation for the n-ary families exhibits the familiar pattern: existential closure of a conjunctive statement for the product, universal closure of disjunctions for the residual implications.

$$
\begin{aligned}
v(f_{\bullet}(A_1, \ldots, A_n)) &= \{x \mid \exists y_1 \ldots y_n (Rx, y_1 \ldots y_n \ \& \ y_1 \in v(A_1) \ \& \ \ldots \& \ y_n \in v(A_n)\} \\
v(f^{i}_{\rightarrow}(A_1, \ldots, A_n)) &= \{x \mid \forall y_1 \ldots y_n ((R^{-i} x, y_1 \ldots y_n \ \& \ y_{j(j \neq i)} \in v(A_j)) \Rightarrow y_i \in v(A_i)\}
\end{aligned}
$$

Unary residuated pairs

Let us focus now on the case of *unary* connectives. The connectives $\diamond$, $\Box^{\downarrow}$ form (dual) residuated pairs (analogous to future possibility, past necessity), interpreted with respect to a binary accessibility relation R^2. One can think of $\Diamond$ ($\Box^\downarrow$) as a truncated multiplicative product (implication). The defining residuation inference ($\star$) here takes the form ($\star\star$).

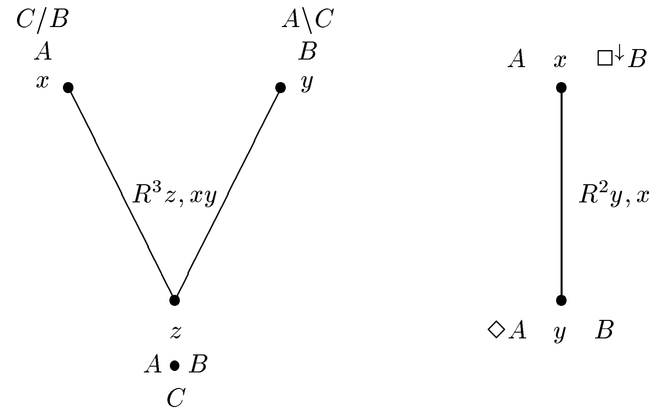
$$\begin{array}{lccc} (\star) & fx \leq y & \Longleftrightarrow & x \leq gy \\ (\star\star) & \Diamond A \to B & \Longleftrightarrow & A \to \Box^\downarrow B \end{array}$$

Again the extended valuation for $\Diamond, \Box^\downarrow$ formulae has the required properties for residuation to arise: existential closure of a conjunctive statement for $\Diamond$, universal closure of a disjunction for the residual $\Box^\downarrow$. Note carefully that the $\Box^\downarrow$ interpretation moves you *back* along the R^2 accessibility relation.

$$\begin{array}{r} v(\Diamond A) = \{x \mid \exists y(Rxy \;\wedge\; y \in v(A)\} \\ v(\Box^\downarrow A) = \{x \mid \forall y(Ryx \;\Rightarrow y \in v(A)\} \end{array}$$

Kripke graphs: unary vs binary connectives

A picture may clarify the relation between the unary and the binary residuated pairs of connectives. In the case of $\bullet$ we make an existential move along the branching accessibility relation R^3. In the case of $\Diamond$ we make an existential move in the same direction, this time for a non-branching accessibility relation. In both cases, universal moves in the opposite direction bring you back to the point of origin.



$$A \to C/B \Leftrightarrow A \bullet B \to C \qquad\qquad \Diamond A \to B \Leftrightarrow A \to \Box^\downarrow B$$
$$A \bullet B \to C \Leftrightarrow B \to A\backslash C$$

Axiomatisation: Lambek style

Let us put together the binary and the unary families of connectives and consider the mixed language

$$\mathcal{F} ::= \mathcal{A} \mid \mathcal{F}/\mathcal{F} \mid \mathcal{F} \bullet \mathcal{F} \mid \mathcal{F}\backslash\mathcal{F} \mid \Diamond\mathcal{F} \mid \Box^\downarrow\mathcal{F}$$

Axiomatic presentation of the pure logic of residuation for the extended language is given below. We decorate the arrows $A \to B$ with combinator proof terms. This makes it easy later on to refer to deductions by means of their combinator.

$$\mathbf{id}_A : A \to A \qquad \frac{f : A \to B \quad g : B \to C}{g \circ f : A \to C}$$

$$\frac{f : \Diamond A \to B}{\mu(f) : A \to \Box^{\downarrow} B} \qquad \frac{g : A \to \Box^{\downarrow} B}{\mu^{-1}(g) : \Diamond A \to B}$$

$$\frac{f : A \bullet B \to C}{\beta(f) : A \to C/B} \qquad \frac{g : A \to C/B}{\beta^{-1}(g) : A \bullet B \to C}$$

$$\frac{f : A \bullet B \to C}{\gamma(f) : B \to A\backslash C} \qquad \frac{g : B \to A\backslash C}{\gamma^{-1}(g) : A \bullet B \to C}$$

Axiomatisation: Došen style

Below a deductive presentation based on the alternative way of characterizing a pair of residual operations f, g in terms of Isotonicity $(\star)$ and the inequalities $(\star\star)$ for the compositions fg, gf.

$$(\star) \quad x \leq y \Rightarrow fx \leq fy, gx \leq gy$$

$$(\star\star) \quad fgx \leq x, \quad x \leq gfx$$

In this presentation, the **unit**, **co-unit** combinators are primitive type transitions, recursively generalized via the Isotonicity rules of inference (Antitonicity for the negative subtype of implications $/, \backslash$).

$$\mathbf{id}_A : A \to A \qquad \frac{f : A \to B \quad g : B \to C}{g \circ f : A \to C}$$

$$\begin{array}{rr} \mathbf{unit}_{\Box^{\downarrow}} : \Diamond\Box^{\downarrow} A \to A & \mathbf{co\text{-}unit}_{\Box^{\downarrow}} : A \to \Box^{\downarrow}\Diamond A \\ \mathbf{unit}_{/} : A/B \bullet B \to A & \mathbf{co\text{-}unit}_{/} : A \to (A \bullet B)/B \\ \mathbf{unit}_{\backslash} : B \bullet B\backslash A \to A & \mathbf{co\text{-}unit}_{\backslash} : A \to B\backslash(B \bullet A) \end{array}$$

$$\frac{f : A \to B}{(f)^{\Diamond} : \Diamond A \to \Diamond B} \qquad \frac{f : A \to B}{(f)^{\Box^{\downarrow}} : \Box^{\downarrow} A \to \Box^{\downarrow} B}$$

$$\frac{f : A \to B \quad g : C \to D}{f \cdot g : A \bullet C \to B \bullet D}$$

$$\frac{f : A \to B \quad g : C \to D}{f/g : A/D \to B/C} \qquad \frac{f : A \to B \quad g : C \to D}{g\backslash f : D\backslash A \to C\backslash B}$$

Equivalence of the deductive presentations

For the $/, \bullet, \backslash$ fragment, we know the two deductive presentations are equivalent, cf. Lambek [21] for one direction, Došen [9] for the other. We take the Lambek presentation as our starting point here, and show for the extended system how from μ, μ^{-1} we obtain the alternative axiomatisation in terms of Isotonicity and the inequalities for the compositions $\Diamond\Box^{\downarrow}$ and $\Box^{\downarrow}\Diamond$ (Term decoration for the right column left to the reader.)

$$\frac{\mathbf{id}_{\Box^{\downarrow} A} : \Box^{\downarrow} A \to \Box^{\downarrow} A}{\mu^{-1}(\mathbf{id}_{\Box^{\downarrow} A}) : \Diamond\Box^{\downarrow} A \to A} \qquad \frac{\Diamond A \to \Diamond A}{A \to \Box^{\downarrow}\Diamond A}$$

$$\dfrac{f : A \to B \quad \dfrac{\mathbf{id}_{\Diamond B} : \Diamond B \to \Diamond B}{\mu(\mathbf{id}_{\Diamond B}) : B \to \Box^{\downarrow}\Diamond B}}{\dfrac{\mu(\mathbf{id}_{\Diamond B}) \circ f : A \to \Box^{\downarrow}\Diamond B}{\mu^{-1}(\mu(\mathbf{id}_{\Diamond B}) \circ f) : \Diamond A \to \Diamond B}} \qquad \dfrac{\dfrac{\dfrac{\Box^{\downarrow}A \to \Box^{\downarrow}A}{\Diamond\Box^{\downarrow}A \to A} \quad A \to B}{\Diamond\Box^{\downarrow}A \to B}}{\Box^{\downarrow}A \to \Box^{\downarrow}B}$$

Soundness, Completeness

For ternary frame semantics for the $\mathcal{F}(/, \bullet, \backslash)$ fragment, Došen [8] proves soundness and completeness on the basis of a canonical model construction. Došen's results extend unproblematically to the $\Diamond, \Box^{\downarrow}$ extended language. We now consider mixed frames $\langle W, R^2, R^3 \rangle$ with W the set of linguistic resources as before, and R^2, R^3 arbitrary binary and ternary relations on W.

$$\vdash A \to B \text{ iff, for every valuation } v \text{ on every frame, } v(A) \subseteq v(B).$$

($\Rightarrow$) Induction on the length of proofs of $A \to B$. ($\Leftarrow$) Extend Došen's canonical model construction for the R^2 relation as follows. For the canonical frame, let W be the formulae of $\mathcal{F}(/, \bullet, \backslash, \Diamond, \Box^{\downarrow})$. In the canonical frame, we define the accessibility relations R^2 and R^3 as follows:

$$R^3(C, A, B) \iff \vdash C \to A \bullet B \qquad R^2(A, B) \iff \vdash A \to \Diamond B$$

Define the canonical valuation as $v(A) = \{B \mid \ \vdash B \to A\}$. Now suppose $v(A) \subseteq v(B)$ but $\not\vdash A \to B$. If $\not\vdash A \to B$ with the canonical valuation on the canonical frame, $A \in v(A)$ but $A \notin v(B)$ so $v(A) \not\subseteq v(B)$. Contradiction.

Canonical model: compound formulae

We have to check the canonical model construction for the new compound formulae $\Diamond A$, $\Box^{\downarrow}A$. Below the direction that requires a little thinking.

($\Diamond$) Assume $A \in v(\Diamond B)$. We have to show $\vdash A \to \Diamond B$. $A \in v(\Diamond B)$ implies $\exists A'$ such that $R^2 A A'$ and $A' \in v(B)$. By inductive hypothesis, $\vdash A' \to B$. By Isotonicity for $\Diamond$ this implies $\vdash \Diamond A' \to \Diamond B$. We have $\vdash A \to \Diamond A'$ by (Def R^2) in the canonical frame. By Transitivity, $\vdash A \to \Diamond B$.

($\Box^{\downarrow}$) Assume $A \in v(\Box^{\downarrow}B)$. We have to show $\vdash A \to \Box^{\downarrow}B$. $A \in v(\Box^{\downarrow}B)$ implies that $\forall A'$ such that $R^2 A' A$ we have $A' \in v(B)$. Let A' be $\Diamond A$. $R^2 A' A$ holds in the canonical frame since $\vdash \Diamond A \to \Diamond A$. By inductive hypothesis we have $\vdash A' \to B$, i.e. $\vdash \Diamond A \to B$. By Residuation this gives $\vdash A \to \Box^{\downarrow}B$.

Logical versus structural connectives

Following the agenda set out in §1 for the binary connectives, we now introduce a Gentzen presentation, and show that it is equivalent to the deductive presentation. For the Gentzen presentation we prove Cut Elimination, with its pleasant corollaries of Decidability and the Subformula property.

In order to present Gentzen calculus for the extended type language, we need an n-ary structural operator for every family of n-ary logical operators: binary $(\cdot, \cdot)$ for the family $/, \bullet, \backslash$, and unary $(\cdot)$ for the family $\Diamond, \Box^{\downarrow}$. Corresponding to the formula language $\mathcal{F}$ we have a language of terms $\mathcal{T}$ (structured configurations of formulae).

$$\mathcal{F} ::= \mathcal{A} \mid \mathcal{F}/\mathcal{F} \mid \mathcal{F} \bullet \mathcal{F} \mid \mathcal{F}\backslash\mathcal{F} \mid \Diamond\mathcal{F} \mid \Box^{\downarrow}\mathcal{F}$$

$$\mathcal{T} ::= \mathcal{F} \mid (\mathcal{T}, \mathcal{T}) \mid (\mathcal{T})$$

Gentzen presentation

As before, sequents are pairs (Γ, A), $\Gamma \in \mathcal{T}$, $A \in \mathcal{F}$, written $\Gamma \Rightarrow A$. We have Belnap-style antecedent punctuation, with for $\Diamond, \Box^{\downarrow}$ the *unary* structural connective $(\cdot)$ matching the unary logical connectives. Below the rules of use $[\Diamond L]$, $[\Box^{\downarrow} L]$ and the rules of proof $[\Diamond R]$, $[\Box^{\downarrow} R]$ for the new connectives. It may be helpful to compare the $\Diamond$ rules with the rules for $\bullet$, and the $\Box^{\downarrow}$ rules with the rules for an implication, say $/$.

Compare the $\Diamond$ rules with the rules for binary product $\bullet$:

$$\frac{\Gamma \Rightarrow A}{(\Gamma) \Rightarrow \Diamond A}\ \Diamond R \qquad \frac{\Gamma[(A)] \Rightarrow B}{\Gamma[\Diamond A] \Rightarrow B}\ \Diamond L$$

$$\frac{\Gamma \Rightarrow A \quad \Delta \Rightarrow B}{(\Gamma, \Delta) \Rightarrow A \bullet B}\ \bullet R \qquad \frac{\Gamma[(A, B)] \Rightarrow C}{\Gamma[A \bullet B] \Rightarrow C}\ \bullet L$$

Compare again the $\Box^{\downarrow}$ rules with the rules for binary implication $/$:

$$\frac{(\Gamma) \Rightarrow A}{\Gamma \Rightarrow \Box^{\downarrow} A}\ \Box^{\downarrow} R \qquad \frac{\Gamma[A] \Rightarrow B}{\Gamma[(\Box^{\downarrow} A)] \Rightarrow B}\ \Box^{\downarrow} L$$

$$\frac{(\Gamma, B) \Rightarrow A}{\Gamma \Rightarrow A/B}\ /R \qquad \frac{\Delta \Rightarrow B \quad \Gamma[A] \Rightarrow C}{\Gamma[(A/B, \Delta)] \Rightarrow C}\ /L$$

The astute reader will observe that, with some adaptation of the notation, the $[\Diamond L]$, $[\Diamond R]$ and $[\Box^{\downarrow} R]$ rules above are the ones Morrill ([27]) proposes in (one of the versions) of his 'bracket' operators. (The $[\Box^{\downarrow} L]$ rule of [27] is flawed: it derives the non-theorem $\Box^{\downarrow}\Diamond A \Rightarrow A$.) But the same observant reader will notice that the intended semantics for the bracket operators in [27] and [28] is less general, in assuming (quasi)functionality for the bracket/anti-bracket operators. Because our objective here is to start from the pure logic of residuation for $\Diamond, \Box^{\downarrow}$, we stick to the unconstrained relational semantics for R^2 for which completeness holds.

Extending the valuation to structured terms

To obtain direct semantic interpretation for sequents $\Gamma \Rightarrow A$ we can extend the valuation to antecedent terms. Compare again the valuation for logical operators $\bullet$, $\Diamond$ with that of their structural counterparts $(\cdot, \cdot)$, $(\cdot)$

$$\begin{array}{lcl} v(A \bullet B) & = & \{z \mid \exists x \exists y [Rz, xy \ \&\ x \in v(A) \ \&\ y \in v(B)]\} \\ v((\Delta_1, \Delta_2)) & = & \{z \mid \exists x \exists y [Rz, xy \ \&\ x \in v(\Delta_1) \ \&\ y \in v(\Delta_2)]\} \end{array}$$

$$\begin{array}{lcl} v(\Diamond A) & = & \{x \mid \exists y (Rxy \ \wedge \ y \in v(A)\} \\ v((\Delta)) & = & \{x \mid \exists y (Rxy \ \wedge \ y \in v(\Delta)\} \end{array}$$

Truth and validity for sequents are then defined the usual way. We have completeness in case $\vdash \Gamma \Rightarrow A$ iff for every valuation v on every frame we have $v(\Gamma) \subseteq v(A)$.

Equivalence of Gentzen and combinator presentations

To compare the two presentations, define for the language $\mathcal{F}(/, \bullet, \backslash, \Diamond, \Box^{\downarrow})$ the formula representation $\Delta^{\flat}$ of a structural configuration Δ as follows:

$$(\Delta_1, \Delta_2)^{\flat} = \Delta_1^{\flat} \bullet \Delta_2^{\flat}, \quad (\Delta)^{\flat} = \Diamond \Delta^{\flat}, \quad A^{\flat} = A$$

The sequent presentation for the language $\mathcal{F}(/, \bullet, \backslash, \Diamond, \Box^{\downarrow})$ can be shown to be equivalent to the combinator axiomatisation in the sense of the following proposition.

Every combinator $f : A \to B$ gives a proof of $A \Rightarrow B$, and every proof of a sequent $\Gamma \Rightarrow B$ gives a combinator $f : \Gamma^\flat \to B$.

From combinators to sequents

To obtain the Gentzen rules [$\Diamond$L], [$\Diamond$R], [$\Box^\downarrow$L], [$\Box^\downarrow$R] from combinator deductions, we use Isotonicity of $\Diamond, \Box^\downarrow$ in addition to the residuation inferences μ, μ^{-1}. Given the formula equivalent $\Gamma^\flat$ for sequent terms Γ, μ gives $[\Box^\downarrow\mathrm{R}]^\flat$, $[\Diamond\mathrm{L}]^\flat$ makes premise and conclusion identical, and Isotonicity for $\Diamond$ gives $[\Diamond\mathrm{R}]^\flat$. The only non-trivial case is [$\Box^\downarrow$L]. Consider first the case where the context Γ is empty. The combinator derivation of $[\Box^\downarrow\mathrm{L}]^\flat$ is given below.

$$\dfrac{\dfrac{f : A \to B}{(f)^{\Box^\downarrow} : \Box^\downarrow A \to \Box^\downarrow B}}{\mu((f)^{\Box^\downarrow}) : \Diamond\Box^\downarrow A \to B} \qquad\qquad \dfrac{A \Rightarrow B}{(\Box^\downarrow A) \Rightarrow B}\ \Box^\downarrow L$$

Next the case where the context Γ in the [$\Box^\downarrow$L] premise $\Gamma[A] \Rightarrow B$ is non-empty. Let g be $\Gamma[A]^\flat \to B$. Let $\pi(g)$ be a sequence of μ, β, γ residuation inferences isolating A on the left of the arrow. Then we obtain the formula equivalent of the conclusion of [$\Box^\downarrow$L] via the deduction $\pi^{-1}(\mu(\Box^\downarrow(\pi(g))))$.

From sequents to combinators

To obtain the combinators **id**, $f \circ g$ (Transitivity), $\mu(f)$, $\mu^{-1}(g)$ (Residuation) from sequent derivations, we use the Cut rule. Once we have established the equivalence of the combinator and the sequent presentation, we prove Cut Elimination for the latter. [Ax] gives **id**, $f \circ g$ is a special case of Cut. The crucial new cases $\mu(f)$, $\mu^{-1}(g)$ follow.

$$\dfrac{\dfrac{\dfrac{\dfrac{A \Rightarrow A}{(A) \Rightarrow \Diamond A}\ \Diamond R}{A \Rightarrow \Box^\downarrow\Diamond A}\ \Box^\downarrow R \quad \dfrac{f : \Diamond A \Rightarrow B}{(\Box^\downarrow\Diamond A) \Rightarrow B}\ \Box^\downarrow L}{(A) \Rightarrow B}\ (cut)}{\mu(f) : A \Rightarrow \Box^\downarrow B}\ \Box^\downarrow R \qquad \dfrac{\dfrac{g : A \Rightarrow \Box^\downarrow B \quad \dfrac{B \Rightarrow B}{(\Box^\downarrow B) \Rightarrow B}\ \Box^\downarrow L}{(A) \Rightarrow B}\ (cut)}{\mu^{-1}(g) : \Diamond A \Rightarrow B}\ \Diamond L$$

Cut elimination: principal cuts

We now extend the Cut Elimination result to the new connectives $\Diamond, \Box^\downarrow$. We proceed by induction on the complexity of the cut formula, and distinguish principal cuts, where the cut formula is active in both cut premises, from permutation conversions, where this is not the case.

Below the new cases of principal cuts, with cut formula $\Diamond A$ and $\Box^\downarrow A$. Replacement of a cut on $\Diamond A$ ($\Box^\downarrow A$) by a cut on A of smaller degree.

$$\dfrac{\dfrac{\Delta \Rightarrow A}{(\Delta) \Rightarrow \Diamond A}\ \Diamond R \quad \dfrac{\Gamma[(A)] \Rightarrow B}{\Gamma[\Diamond A] \Rightarrow B}\ \Diamond L}{\Gamma[(\Delta)] \Rightarrow B}\ (cut) \quad \rightsquigarrow \quad \dfrac{\Delta \Rightarrow A \quad \Gamma[(A)] \Rightarrow B}{\Gamma[(\Delta)] \Rightarrow B}\ (cut)$$

$$\dfrac{\dfrac{(\Delta) \Rightarrow A}{\Delta \Rightarrow \Box^\downarrow A}\ \Box^\downarrow R \quad \dfrac{\Gamma[A] \Rightarrow B}{\Gamma[(\Box^\downarrow A)] \Rightarrow B}\ \Box^\downarrow L}{\Gamma[(\Delta)] \Rightarrow B}\ (cut) \quad \rightsquigarrow \quad \dfrac{(\Delta) \Rightarrow A \quad \Gamma[A] \Rightarrow B}{\Gamma[(\Delta)] \Rightarrow B}\ (cut)$$

Cut elimination: permutation conversions

The new cases where the active formula in the left or right premise is different from the Cut formula allow for the usual elimination strategy: permutation of the Cut rule and the logical rule. The Cut is moved upwards , and becomes of lower degree. Below the left premise antecedent cases for $\diamond A$ and $\Box^{\downarrow} A$.

$$\dfrac{\dfrac{\Gamma[(A)] \Rightarrow B}{\Gamma[\diamond A] \Rightarrow B}\diamond L \quad \Delta[B] \Rightarrow C}{\Delta[\Gamma[\diamond A]] \Rightarrow C}(cut) \quad \rightsquigarrow \quad \dfrac{\dfrac{\Gamma[(A)] \Rightarrow B \quad \Delta[B] \Rightarrow C}{\Delta[\Gamma[(A)]] \Rightarrow C}(cut)}{\Delta[\Gamma[\diamond A]] \Rightarrow C}\diamond L$$

$$\dfrac{\dfrac{\Delta[A] \Rightarrow B}{\Delta[(\Box^{\downarrow} A)] \Rightarrow B}\Box^{\downarrow} L \quad \Gamma[B] \Rightarrow C}{\Gamma[\Delta[(\Box^{\downarrow} A)]] \Rightarrow C}(cut) \quad \rightsquigarrow \quad \dfrac{\dfrac{\Delta[A] \Rightarrow B \quad \Gamma[B] \Rightarrow C}{\Gamma[\Delta[A]] \Rightarrow C}(cut)}{\Gamma[\Delta[(\Box^{\downarrow} A)]] \Rightarrow C}\Box^{\downarrow} L$$

Permutation conversion: right premise antecedent

Active type $\diamond A$ or $\Box^{\downarrow} A$ in the antecedent of the right Cut premise. Notation: $\Gamma[\Delta_1, \Delta_2]$ for a structure Γ with substructures Δ_1, Δ_2, not necessarily sisters.

$$\dfrac{\Delta \Rightarrow A \quad \dfrac{\Gamma[A, (B)] \Rightarrow C}{\Gamma[A, \diamond B] \Rightarrow C}\diamond L}{\Gamma[\Delta, \diamond B] \Rightarrow C}(cut) \quad \rightsquigarrow \quad \dfrac{\dfrac{\Delta \Rightarrow A \quad \Gamma[A, (B)] \Rightarrow C}{\Gamma[\Delta, (B)] \Rightarrow C}(cut)}{\Gamma[\Delta, \diamond B] \Rightarrow C}\diamond L$$

$$\dfrac{\Delta \Rightarrow A \quad \dfrac{\Gamma[A, B] \Rightarrow C}{\Gamma[A, (\Box^{\downarrow} B)] \Rightarrow C}\Box^{\downarrow} L}{\Gamma[\Delta, (\Box^{\downarrow} B)] \Rightarrow C}(cut) \quad \rightsquigarrow \quad \dfrac{\dfrac{\Delta \Rightarrow A \quad \Gamma[A, B] \Rightarrow C}{\Gamma[\Delta, B] \Rightarrow C}(cut)}{\Gamma[\Delta, (\Box^{\downarrow} B)] \Rightarrow C}\Box^{\downarrow} L$$

Permutation conversion: right premise succedent

Active type $\diamond A$ or $\Box^{\downarrow} A$ in the succedent of the right Cut premise.

$$\dfrac{\Gamma \Rightarrow A \quad \dfrac{(\Delta[A]) \Rightarrow B}{\Delta[A] \Rightarrow \Box^{\downarrow} B}\Box^{\downarrow} R}{\Delta[\Gamma] \Rightarrow \Box^{\downarrow} B}(cut) \quad \rightsquigarrow \quad \dfrac{\dfrac{\Gamma \Rightarrow A \quad (\Delta[A]) \Rightarrow B}{(\Delta[\Gamma]) \Rightarrow B}(cut)}{\Delta[\Gamma] \Rightarrow \Box^{\downarrow} B}\Box^{\downarrow} R$$

$$\dfrac{\Delta \Rightarrow A \quad \dfrac{\Gamma[A] \Rightarrow B}{(\Gamma[A]) \Rightarrow \diamond B}\diamond R}{(\Gamma[\Delta]) \Rightarrow \diamond B}(cut) \quad \rightsquigarrow \quad \dfrac{\dfrac{\Delta \Rightarrow A \quad \Gamma[A] \Rightarrow B}{\Gamma[\Delta] \Rightarrow B}(cut)}{(\Gamma[\Delta]) \Rightarrow \diamond B}\diamond R$$

Illustration: Residuation laws

As an example, we check the compositions $\diamond\Box^{\downarrow}$ and $\Box^{\downarrow}\diamond$ (cf Application, Co-Application). Below their cut-free Gentzen derivations.

$$\dfrac{\dfrac{A \Rightarrow A}{(\Box^{\downarrow} A) \Rightarrow A}\Box^{\downarrow} L}{\diamond\Box^{\downarrow} A \Rightarrow A}\diamond L \qquad \dfrac{\dfrac{A \Rightarrow A}{(A) \Rightarrow \diamond A}\diamond R}{A \Rightarrow \Box^{\downarrow}\diamond A}\Box^{\downarrow} R$$

$$fgx \leq x \qquad\qquad x \leq gfx$$

Structural postulates

What we have discussed so far is the pure logic of residuation for the unary family $\diamond, \Box^{\downarrow}$. By imposing conditions ASS, COMM or their combination on ternary frames, we generate the landscape **NL, L, NLP, LP** with completeness results for the relevant classes of frames (Došen). Along the same lines, we can develop the substructural landscape for the unary family $\diamond, \Box^{\downarrow}$ and its binary accessibility relation R^2, and for the mixed R^2, R^3 system.

The following structural postulates constrain R^2 to be transitive (4), or reflexive (T). Communication between R^2 and R^3 can be established via the strong distributivity postulate K, which distributes unary $\diamond$ over both components of a binary $\bullet$, or, in a more constrained way, via the weak distributivity postulates $K1$, $K2$, where $\diamond$ selects the left or right subtype of a product.

$$\begin{array}{rc} 4: & \diamond\diamond A \to \diamond A \\ T: & A \to \diamond A \\ K1: & \diamond(A \bullet B) \to \diamond A \bullet B \\ K2: & \diamond(A \bullet B) \to A \bullet \diamond B \\ K: & \diamond(A \bullet B) \to \diamond A \bullet \diamond B \end{array}$$

Below the correponding frame conditions ($\forall x, y, z, w \in W$).

$$\begin{array}{rc} 4: & (Rxy \ \& \ Ryz) \Rightarrow Rxz \\ T: & Rxx \\ K(1,2): & (Rwx \ \& \ Rxyz) \Rightarrow \exists y'(Ry'y \ \& \ Rwy'z) \ \vee \ \exists z'(Rz'z \ \& \ Rwyz') \\ K: & (Rwx \ \& \ Rxyz) \Rightarrow \exists y' \exists z'(Ry'y \ \& \ Rz'z \ \& \ Rwy'z') \end{array}$$

The K condition is the correlation postulate for Relevance logic from Routley and Meyer [31]. See Kurtonina [20] for discussion in the context of logics of linguistic resources. The weak distributivity principles $K1, K2$ play an important role in the applications discussed later in this paper.

Structural rules

The structural rules below translate the postulates $T, 4, K1, K2, K$ from the formula level to the term level. As before, we prove equivalence between the rule and the postulate versions, and show that the Gentzen formulation allows cut-free proof search.

$$\frac{\Gamma[(\Delta)] \Rightarrow A}{\Gamma[((\Delta))] \Rightarrow A}\ 4 \qquad \frac{\Gamma[(\Delta)] \Rightarrow A}{\Gamma[\Delta] \Rightarrow A}\ T$$

$$\frac{\Gamma[((\Delta_1), \Delta_2)] \Rightarrow A}{\Gamma[((\Delta_1, \Delta_2))] \Rightarrow A}\ K1 \quad \frac{\Gamma[((\Delta_1), (\Delta_2))] \Rightarrow A}{\Gamma[((\Delta_1, \Delta_2))] \Rightarrow A}\ K \quad \frac{\Gamma[(\Delta_1, (\Delta_2))] \Rightarrow A}{\Gamma[((\Delta_1, \Delta_2))] \Rightarrow A}\ K2$$

Structural rules from structural postulates

We have to extend the equivalence between axiomatic and Gentzen style presentation to the structural postulates and rules. To obtain the sequent rules $T, 4, K$ from combinator deductions, it is enough to consider the case where the context Γ is empty, as we have seen above. The following deductions give the formula equivalent of the structural rules $T, 4, K$. We leave $K1, K2$ to the reader.

$$\frac{4 : \Diamond\Diamond\Delta^\flat \to \Diamond\Delta^\flat \quad f : \Diamond\Delta^\flat \to A}{f \circ 4 : \Diamond\Diamond\Delta^\flat \to A} \quad \rightsquigarrow \quad \frac{(\Delta)^\flat \Rightarrow A}{((\Delta))^\flat \Rightarrow A}\ 4$$

$$\frac{T : \Delta^\flat \to \Diamond\Delta^\flat \quad f : \Diamond\Delta^\flat \to A}{f \circ T : \Delta^\flat \to A} \quad \rightsquigarrow \quad \frac{(\Delta)^\flat \Rightarrow A}{\Delta^\flat \Rightarrow A}\ T$$

$$\frac{K : \Diamond(\Delta_1^\flat \bullet \Delta_2^\flat) \to \Diamond\Delta_1^\flat \bullet \Diamond\Delta_2^\flat \quad f : \Diamond\Delta_1^\flat \bullet \Diamond\Delta_2^\flat \to A}{f \circ K : \Diamond(\Delta_1^\flat \bullet \Delta_2^\flat) \to A} \quad \rightsquigarrow \quad \frac{((\Delta_1),(\Delta_2))^\flat \Rightarrow A}{((\Delta_1,\Delta_2))^\flat \Rightarrow A}\ K$$

Structural postulates from structural rules

Derivation of the structural postulates via Gentzen proofs is straightforward.

$$\dfrac{\dfrac{A \Rightarrow A}{(A) \Rightarrow \Diamond A}\ \Diamond R}{A \Rightarrow \Diamond A}\ T \qquad \dfrac{\dfrac{\dfrac{\dfrac{A \Rightarrow A}{(A) \Rightarrow \Diamond A}\ \Diamond R}{((A)) \Rightarrow \Diamond A}\ 4}{(\Diamond A) \Rightarrow \Diamond A}\ \Diamond L}{\Diamond\Diamond A \Rightarrow \Diamond A}\ \Diamond L \qquad \dfrac{\dfrac{\dfrac{\dfrac{\dfrac{A \Rightarrow A}{(A) \Rightarrow \Diamond A}\ \Diamond R \quad \dfrac{B \Rightarrow B}{(B) \Rightarrow \Diamond B}\ \Diamond R}{((A),(B)) \Rightarrow \Diamond A \bullet \Diamond B}\ \bullet R}{((A,B)) \Rightarrow \Diamond A \bullet \Diamond B}\ K}{((A \bullet B)) \Rightarrow \Diamond A \bullet \Diamond B}\ \bullet L}{\Diamond(A \bullet B) \Rightarrow \Diamond A \bullet \Diamond B}\ \Diamond L$$

Cut elimination: structural rules

We extend the cut elimination algorithm to logics with a structural rule package from $\mathcal{P}(\{T, 4, K\})$. Recall that in the case of connectives the proof of the Cut Elimination theorem is by induction on the complexity of Cut inferences, measured in terms of the number of connectives in the cut formula. The structural rules do not involve decomposition of formulae, so we need an additional complexity measure here.

Following [9, 3], let the *trace* of a cut formula A be the sum of the lengths of the paths in the derivations of the cut premises connecting the two occurrences of A with the point of their first introduction in the proof. The cut elimination steps involving structural rules now assimilate to the permutation cases: if a structural rule feeds the cut inference, we can interchange the order of application of the cut and the structural rule, leading to a situation with decreased *trace*, as the inductive hypothesis requires. Two examples are given below.

$$\dfrac{\dfrac{\Delta[(\Delta')] \Rightarrow A}{\Delta[\Delta'] \Rightarrow A}\ T \quad \Gamma[A] \Rightarrow B}{\Gamma[\Delta[\Delta']] \Rightarrow B}\ (cut) \quad \rightsquigarrow \quad \dfrac{\dfrac{\Delta[(\Delta')] \Rightarrow A \quad \Gamma[A] \Rightarrow B}{\Gamma[\Delta[(\Delta')]] \Rightarrow B}\ (cut)}{\Gamma[\Delta[\Delta']] \Rightarrow B}\ T$$

$$\dfrac{\Delta \Rightarrow A \quad \dfrac{\Gamma[(\Gamma'[A])] \Rightarrow B}{\Gamma[((\Gamma'[A]))] \Rightarrow B}\ 4}{\Gamma[((\Gamma'[\Delta]))] \Rightarrow B}\ (cut) \quad \rightsquigarrow \quad \dfrac{\dfrac{\Delta \Rightarrow A \quad \Gamma[(\Gamma'[A])] \Rightarrow B}{\Gamma[(\Gamma'[\Delta])] \Rightarrow B}\ (cut)}{\Gamma[((\Gamma'[\Delta]))] \Rightarrow B}\ 4$$

Structural postulates: universal variant

In our discussion of structural postulates for $\bullet$, we have seen that we can express Associativity, Commutativity either via a $\bullet$ postulate, or via implicational postulates, if we prefer to keep the language product-free. In a similar vein we could have presented $T, 4, K$ in their $\Box^{\downarrow}$ forms:

$$\begin{array}{rc} 4\Box^{\downarrow}: & \Box^{\downarrow}A \rightarrow \Box^{\downarrow}\Box^{\downarrow}A \\ T\Box^{\downarrow}: & \Box^{\downarrow}A \rightarrow A \\ K\Box^{\downarrow}/: & \Box^{\downarrow}(A/B) \rightarrow \Box^{\downarrow}A/\Box^{\downarrow}B \\ K\Box^{\downarrow}\backslash: & \Box^{\downarrow}(B\backslash A) \rightarrow \Box^{\downarrow}B\backslash\Box^{\downarrow}A \end{array}$$

Below an illustration for the derivation of the universal variant $K\Box^{\downarrow}/$.

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{B \Rightarrow B \quad A \Rightarrow A}{(A/B, B) \Rightarrow A}\ /L}{((\Box^{\downarrow}(A/B)), (\Box^{\downarrow}B)) \Rightarrow A}\ \Box^{\downarrow}L, \Box^{\downarrow}L}{((\Box^{\downarrow}(A/B), \Box^{\downarrow}B)) \Rightarrow A}\ K}{(\Box^{\downarrow}(A/B), \Box^{\downarrow}B) \Rightarrow \Box^{\downarrow}A}\ \Box^{\downarrow}R}{\Box^{\downarrow}(A/B) \Rightarrow \Box^{\downarrow}A/\Box^{\downarrow}B}\ /R$$

S4: Compilation of structural rules

We saw above that in the presence of Associativity for $\bullet$, we have a sugared Gentzen presentation where the structural rule is compiled away, and the binary sequent punctuation $(\cdot, \cdot)$ omitted. Analogously, for $\Box^{\downarrow}$ with the combination $KT4$ ($S4$), we have a sugared version of the Gentzen calculus, where the $KT4$ structural rules are compiled away, so that the unary $(\cdot)$ punctuation can be omitted. Compare the following. (Notation $\dagger\Gamma$ for a term Γ of which the (pre)terminal subterms are of the form $\dagger A$. The $4(cut)$ step is a series of replacements of terminal $\Box^{\downarrow}A$ by $\Box^{\downarrow}\Box^{\downarrow}A$ via cuts depending on 4.)

$$\dfrac{\dfrac{\Gamma[A] \Rightarrow B}{\Gamma[(\Box^{\downarrow}A)] \Rightarrow B}\ \Box^{\downarrow}L}{\Gamma[\Box^{\downarrow}A] \Rightarrow B}\ T \quad \rightsquigarrow \quad \dfrac{\Gamma[A] \Rightarrow B}{\Gamma[\Box^{\downarrow}A] \Rightarrow B}\ \Box^{\downarrow}L(S4)$$

$$\dfrac{\dfrac{\dfrac{\dfrac{\Box^{\downarrow}\Gamma \Rightarrow A}{(\Box^{\downarrow}\Box^{\downarrow})\Gamma \Rightarrow A}\ \Box^{\downarrow}L}{(\Box^{\downarrow})\Gamma \Rightarrow A}\ 4(cut)}{(\Box^{\downarrow}\Gamma) \Rightarrow A}\ K}{\Box^{\downarrow}\Gamma \Rightarrow \Box^{\downarrow}A}\ \Box^{\downarrow}R \quad \rightsquigarrow \quad \dfrac{\Box^{\downarrow}\Gamma \Rightarrow A}{\Box^{\downarrow}\Gamma \Rightarrow \Box^{\downarrow}A}\ \Box^{\downarrow}R(S4)$$

Embedding NL in modal L: locality constraints

Morrill ([28]) conjectures the following. Define an embedding translation $(\cdot)^{\sharp}$: $\mathcal{F}(\mathbf{NL}) \mapsto \mathcal{F}(\mathbf{L} + \{\Diamond, \Box^{\downarrow}\})$:

$$\begin{array}{c} (p)^{\sharp} = p \\ (A \bullet B)^{\sharp} = \Diamond((A)^{\sharp} \bullet (B)^{\sharp}) \\ (A/B)^{\sharp} = \Box^{\downarrow}(A)^{\sharp}/(B)^{\sharp} \\ (B\backslash A)^{\sharp} = (B)^{\sharp}\backslash\Box^{\downarrow}(A)^{\sharp} \end{array}$$

$$\mathbf{NL} \vdash A \rightarrow B \text{ iff } \mathbf{L}{+}\{\Diamond, \Box^{\downarrow}\} \vdash (A)^{\sharp} \rightarrow (B)^{\sharp}$$

A (non-trivial) proof of the conjecture, based on a semantic completeness argument, is given in Kurtonina [20]. The linguistic relevance is the following: Hepple in [16] proposes a treatment of locality constraints in terms of an $S4$ universal modality. A legitimate question, from our 'minimalistic' perspective is: What resource management properties are needed to enforce a general theory of locality constraints? Proof of the Morrill conjecture above shows that in fact the pure logic of residuation for the pair $\Diamond, \Box^{\downarrow}$ is enough.

Application: modalities as procedural control features

There is a wide range of linguistic applications for the modal operators $\Diamond, \Box^{\downarrow}$. For a start, we can now 'situate' with a high degree of precision the various unary operators that have been proposed in the literature. At one end of the spectrum, one finds a trimmed-down version of Morrill's bracket operators realizing the pure logic of residuation for $\Diamond, \Box^{\downarrow}$. At the other end, we find Hepple's $KT4$ syntactic domain modality. And — more excitingly — one can look for occupants of corners of the landscape that have not been explored. Here we give one surprising application, from the field of algorithmic proof theory. Below we show how one can enforce a uniform head-driven search regime for **L** on the basis of a modal decoration of **L** sequents.

Uniform head-driven search: L*

In the literature on automated deduction, it is well known that cut-free Gentzen proof search is still suboptimal from the efficiency perspective: there may be different (cut-free!) derivations leading to one and the same proof term. Restricting ourselves to the implicational fragment, the spurious non-determinism in the search space has two causes ([35]): (i) permutability of [L] and [R] inferences, and (ii) permutability of [L] inferences among themselves, i.e. non-determinism in the choice of the active formula in the antecedent. A so-called *goal directed* (or: uniform) search regime performs the non-branching [R] inferences before the [L] inferences (re (i)), whereas *head driven* search commits the choice of the antecedent active formula in terms of the goal formula (re (ii)).

In the context of categorial theorem proving, a goal-directed head-driven regime for **L** has been proposed in Hepple [16] with a proof of the safeness (no proof terms are lost) and non-redundancy (each proof term has a unique derivation). We present the Hepple regime in the format of Hendriks [15] where the reader can find a detailed discussion of the spurious ambiguity problem in the context of Gentzen proof search.

$$[\mathrm{Ax}/\star\mathrm{L}]\frac{}{x : p^\star \Rightarrow x : p} \qquad \frac{\Gamma, u : B^\star, \Gamma' \Rightarrow t : p}{\Gamma, u : B, \Gamma' \Rightarrow t : p^\star}[\star\mathrm{R}]$$

$$[/\mathrm{R}]\frac{\Delta, x : B \Rightarrow t : A^\star}{\Delta \Rightarrow \lambda x.t : A/B^\star} \qquad \frac{\Delta \Rightarrow u : B^\star \quad \Gamma, x : A^\star, \Gamma' \Rightarrow t : C}{\Gamma, s : A/B^\star, \Delta, \Gamma' \Rightarrow t[x/su] : C}[/\mathrm{L}]$$

$$[\backslash\mathrm{R}]\frac{x : B, \Delta \Rightarrow t : A^\star}{\Delta \Rightarrow \lambda x.t : B\backslash A^\star} \qquad \frac{\Delta \Rightarrow u : B^\star \quad \Gamma, x : A^\star, \Gamma' \Rightarrow t : C}{\Gamma, \Delta, s : B\backslash A^\star, \Gamma' \Rightarrow t[x/su] : C}[\backslash\mathrm{L}]$$

Comments

The **L*** calculus eliminates the spurious non-determinism of the original presentation **L** by annotating sequents with a procedural control operator '*'. Goal sequents $\Gamma \Rightarrow t : A$ in **L** are replaced by **L*** goal sequents $\Gamma \Rightarrow t : A^\star$. With respect to the first cause of spurious ambiguity (permutability of [L] and [R] inferences), the control part of the [R] inferences forces one to remove all connectives from the succedent until one reaches an atomic succedent.

At that point, the '*' control is transmitted from succedent to antecedent: the [*R] selects an active antecedent formula the head of which ultimately, by force of the control version of the Axiom sequent [*L], will have to match the (now atomic) goal type. The [L] implication inferences initiate a '*' control derivation on the minor premise, and transmit the '*' active declaration from conclusion to major (right) premise. The effect of the flow of control information is to commit the search to the target type selected in the [*R] step. This removes the second source of spurious ambiguity: permutability of [L] inferences.

Our labeling for the '*' version of the Axiom sequent, [*L], is suggestive for the type of modal control we are about to propose.

Proofs and readings

The following theorem from [15] sums up the situation with respect to proofs and readings in **L** and **L***.

1. $\mathbf{L}^\star \vdash \Gamma \Rightarrow A^\star$ iff $\mathbf{L} \vdash \Gamma \Rightarrow A$
2. $\mathbf{L}^\star \vdash \Gamma \Rightarrow t : A^\star$ implies $\mathbf{L} \vdash \Gamma \Rightarrow t : A$
3. $\mathbf{L} \vdash \Gamma \Rightarrow t : A$ implies $\exists t', t' = t$ and $\mathbf{L}^\star \vdash \Gamma \Rightarrow t' : A^\star$
4. if π_1 is an **L*** proof of $\Gamma \Rightarrow t : A$ and π_2 is an **L*** proof of $\Gamma \Rightarrow t' : A$ with $t = t'$, then $\pi_1 = \pi_2$

Of the above, (1) asserts that, syntactically, derivability in **L** and **L*** coincide. Semantically, the set of **L*** proof terms forms a subset of the **L** terms (2). But, modulo logical equivalence, no readings are lost moving from **L** to **L*** (3). Moreover, the **L*** system has the desired one-to-one correspondence between readings and proofs (4).

Uniform proof search: modal control

In this section we show how to enforce the Hepple-Hendriks uniform head-driven search regime via a modal translation. The basic idea is to use the *logical* properties of the connectives $\Diamond, \Box^\downarrow$ to capture the effects of the '*' procedural control marking in **L***. We use the base residuation logic for $\Diamond, \Box^\downarrow$, plus weak distributivity principles $K1, K2$ for the interaction between the unary and the binary families. For convenience we repeat the frame conditions and the Gentzen transformation of the $K1, K2$ structural postulates from our discussion above.

	STRUCTURAL POSTULATE	FRAME CONDITION
$K1:$	$\Diamond(A \bullet B) \rightarrow \Diamond A \bullet B$	$(Rwx \;\&\; Rxyz) \Rightarrow \exists y'(Ry'y \;\&\; Rwy'z)$
$K2:$	$\Diamond(A \bullet B) \rightarrow A \bullet \Diamond B$	$(Rwx \;\&\; Rxyz) \Rightarrow \exists z'(Rz'z \;\&\; Rwyz')$

In structural rule format these take the following form.

$$\frac{\Gamma[((\Delta_1)^\diamond, \Delta_2)] \Rightarrow A}{\Gamma[((\Delta_1, \Delta_2))^\diamond] \Rightarrow A}\, K1 \quad \frac{\Gamma[(\Delta_1, (\Delta_2)^\diamond)] \Rightarrow A}{\Gamma[((\Delta_1, \Delta_2))^\diamond] \Rightarrow A}\, K2$$

To establish the equivalence with **L*** search, we can use the sugared presentation of **L** where Associativity is compiled away so that *binary* punctuation $(\cdot, \cdot)$ can be omitted (but not the unary $(\cdot)^\diamond$!). This gives the following compiled format for $K1, K2$:

$$\frac{\Gamma, (\Delta)^\diamond, \Gamma' \Rightarrow B}{(\Gamma, \Delta, \Gamma')^\diamond \Rightarrow B}\, K'$$

Translation: formulae, sequents

We define the translation mapping first on the formula level, and then extend it to the level of **L*** sequents, where we have to distinguish marked and unmarked formulae. On the formula level, define mappings $(\cdot)^1, (\cdot)^0 : \mathcal{F}(/, \backslash) \mapsto \mathcal{F}(/, \backslash, \diamond, \Box^{\downarrow})$, for antecedent and succedent formula occurrences respectively.

$$\begin{array}{rr} (p)^1 = p & (p)^0 = \Box^{\downarrow} p \\ (A/B)^1 = (A)^1/(B)^0 & (A/B)^0 = (A)^0/\Box^{\downarrow}(B)^1 \\ (B\backslash A)^1 = (B)^0\backslash(A)^1 & (B\backslash A)^0 = \Box^{\downarrow}(B)^1\backslash(A)^0 \end{array}$$

The formulae of a sequent $\Gamma \Rightarrow A$ in **L*** are partitioned by the '*' annotation in a set of marked formulae — a singleton, since there is only one '*' per sequent — and a set of unmarked formulae. We extend the translation mapping taking this difference into account. The antecedent and succedent translation functions $(\cdot)_1, (\cdot)_0$ below are defined in terms of $(\cdot)^1, (\cdot)^0$, but they act in a different way on marked and on unmarked formulae.

$$(A_1, \ldots, A_n)_1 = \overline{A_1}, \ldots, \overline{A_n} \quad \text{where} \quad \overline{A} = \begin{cases} (A)^1 & \text{if } A \text{ is `*' marked} \\ \Box^{\downarrow}(A)^1 & \text{otherwise} \end{cases}$$

$$(A)_0 = \begin{cases} (A)^0 & \text{if } A \text{ is `*' marked} \\ A & \text{otherwise} \end{cases}$$

We now have the following proposition.

$$\mathbf{L}^{\star} \vdash \Gamma \Rightarrow A^{\star} \qquad \text{iff} \qquad \mathbf{L}\diamond\mathbf{K}' \vdash (\Gamma)_1 \Rightarrow (A)_0$$

Equivalence of L* and L$\diamond K'$

The $(\Rightarrow)$ direction of the equivalence can be proved by straightforward induction on the length of derivations in **L***. For the more delicate $(\Leftarrow)$ direction, we have to show that **L**$\diamond K'$ does not derive *more* than **L***. We give a case analysis of the choice points in the top-down (backward-chaining) unfolding of the search space, and show that **L**$\diamond K'$ can make no moves that would lead the search out of the space defined by the translation mapping.

Below we juxtapose the **L*** rules and axiom and their counterpart in **L**$\diamond K'$. We treat only one implication. For the **L**$\diamond K'$ version, we interleave the proof unfolding with the evaluation of the translation mapping. As an auxiliary notion, we have functions ACTIVE and LOCKED which for a sequent return the set of formulae matching the input condition for a logical rule (ACTIVE), and those which no logical rule is applicable to (LOCKED).

Proof search starts with an **L*** goal sequent $\Gamma \Rightarrow (A)^{\star}$. The goal type A is either atomic or complex, the antecedent is of length 1 or greater than 1. Consider first the case of a complex goal type and $1 \leq |\Gamma|$. On the left the **L*** $[/R]$ rule, on the right the corresponding derivation in **L**$\diamond K'$. Both (†) and (‡) stand in a feeding relation with themselves. For the roots of the derivations, we have ACTIVE(†)= $\{A/B\}$, ACTIVE(‡)= $\{(A/B)^0\}$; for the leaves, ACTIVE(†)= $\{A\}$, ACTIVE(‡)= $\{(A)^0\}$. Note that all antecedent formulae in (‡) have main connective $\Box^{\downarrow}$ as a result of the translation mapping. The $\Box^{\downarrow}$ connective acts as a lock: embedded connectives in these formulae will only be accessible after the removal of $\Box^{\downarrow}$.

$$(\dagger) \quad \frac{\Gamma, B \Rightarrow (A)^{\star}}{\Gamma \Rightarrow (A/B)^{\star}}\,/R \qquad \frac{\dfrac{\Box^{\downarrow}(\Gamma)^1, \Box^{\downarrow}(B)^1 \Rightarrow (A)^0}{\Box^{\downarrow}(\Gamma)^1 \Rightarrow (A)^0/\Box^{\downarrow}(B)^1}\,/R}{\Box^{\downarrow}(\Gamma)^1 \Rightarrow (A/B)^0}\,(\cdot)^0 \quad (\ddagger)$$

Consider now the case where the recursion on succedent implications bottoms out, i.e. where we reach the '*' marked atomic head of the goal formula. In **L*** the only applicable rule in this situation is $[\star R]$ which transmits the '*' marking from succedent to antecedent. $[\star R]$ is non-deterministic: any antecedent formula B can receive the '*' marking. In $\mathbf{L}\diamond K'$ the active atom is realized as $(p)^0 = \Box^{\downarrow}p$. The only applicable rule here is $[\Box^{\downarrow}R]$ which, by residuation, realizes $\Box^{\downarrow}$ as $(\cdot)^{\diamond}$ on the antecedent. $[\Box^{\downarrow}R]$ can only be followed by $[K']$, which non-deterministically pushes $(\cdot)^{\diamond}$ to an arbitrary antecedent formula $\Box^{\downarrow}(B)^1$. At that point $[\Box^{\downarrow}L]$ becomes applicable, which through the elimination of $\Box^{\downarrow}$ shifts $(B)^1$ from LOCKED to ACTIVE. Again, roots and leaves of the (†) (‡) derivations agree on ACTIVE and LOCKED.

$$(\dagger)\quad \frac{\Gamma, (B)^{\star}, \Gamma' \Rightarrow p}{\Gamma, B, \Gamma' \Rightarrow (p)^{\star}}\ \star R \qquad \dfrac{\dfrac{\dfrac{\dfrac{\Box^{\downarrow}(\Gamma)^1, (B)^1, \Box^{\downarrow}(\Gamma')^1 \Rightarrow p}{\Box^{\downarrow}(\Gamma)^1, (\Box^{\downarrow}(B)^1)^{\diamond}, \Box^{\downarrow}(\Gamma')^1 \Rightarrow p}\ \Box^{\downarrow}L}{(\Box^{\downarrow}(\Gamma)^1, \Box^{\downarrow}(B)^1, \Box^{\downarrow}(\Gamma')^1)^{\diamond} \Rightarrow p}\ K'}{\Box^{\downarrow}(\Gamma)^1, \Box^{\downarrow}(B)^1, \Box^{\downarrow}(\Gamma')^1 \Rightarrow \Box^{\downarrow}p}\ \Box^{\downarrow}R}{\Box^{\downarrow}(\Gamma)^1, \Box^{\downarrow}(B)^1, \Box^{\downarrow}(\Gamma')^1 \Rightarrow (p)^0}\ (\cdot)^0 \quad (\ddagger)$$

Next we analyse the possible antecedent configurations for the different choices of active formula. The active formula is either atomic or complex, and the context is either empty or non-empty. Let us consider these in turn, starting with the non-empty context case. If the active formula is atomic, the derivation fails, in **L*** and in $\mathbf{L}\diamond K'$. If the active formula is complex (i.e. of the form B/C or $C\backslash B$), the only applicable rule, in **L*** and in $\mathbf{L}\diamond K'$, is $[/L]$ $([\backslash L])$. The derivation branches, initiating uniform head-driven search for the negative subtype of the goal formula in the left premise, and declaring the positive subtype active in the right premise. Roots and leaves of the derivations in **L*** and in $\mathbf{L}\diamond K'$ agree on the ACTIVE, LOCKED partitioning.

$$\frac{\Delta \Rightarrow (B)^{\star} \quad \Gamma, (A)^{\star}, \Gamma' \Rightarrow p}{\Gamma, (A/B)^{\star}, \Delta, \Gamma' \Rightarrow p}\ /L \qquad \dfrac{\dfrac{\Box^{\downarrow}(\Delta)^1 \Rightarrow (B)^0 \quad \Box^{\downarrow}(\Gamma)^1, (A)^1, \Box^{\downarrow}(\Gamma')^1 \Rightarrow p}{\Box^{\downarrow}(\Gamma)^1, (A)^1/(B)^0, \Box^{\downarrow}(\Delta)^1, \Box^{\downarrow}(\Gamma')^1 \Rightarrow p}\ /L}{\Box^{\downarrow}(\Gamma)^1, (A/B)^1, \Box^{\downarrow}(\Delta)^1, \Box^{\downarrow}(\Gamma')^1 \Rightarrow p}\ (\cdot)^1$$

Finally, consider the base cases of the recursion. Below the correspondence when the **L*** Axiom sequent, i.e. $[\star L]$ is reached.

$$\frac{}{(p)^{\star} \Rightarrow p}\ \star L \qquad \frac{p \Rightarrow p}{(p)^1 \Rightarrow p}\ (\cdot)^1$$

For the sake of completeness, one should add the case of the trivial initial sequent $p \Rightarrow (p)^{\star}$, though the issue of spurious ambiguity hardly arises here. Below the **L*** form and its $\mathbf{L}\diamond K'$ counterpart.

$$\dfrac{\dfrac{}{(p)^{\star} \Rightarrow p}\ \star L}{p \Rightarrow (p)^{\star}}\ \star R \qquad \dfrac{\dfrac{\dfrac{p \Rightarrow p}{(\Box^{\downarrow}p)^{\diamond} \Rightarrow p}\ \Box^{\downarrow}L}{\Box^{\downarrow}p \Rightarrow \Box^{\downarrow}p}\ \Box^{\downarrow}R}{\Box^{\downarrow}(p)^1 \Rightarrow (p)^0}\ (\cdot)^1, (\cdot)^0$$

Illustration: Geach

Without the constraint on uniform head-driven search, there are two **L** sequent derivations for the Geach transition. They produce the same proof term.

$$\dfrac{\dfrac{\dfrac{\dfrac{c \Rightarrow c \quad b \Rightarrow b}{b/c, c \Rightarrow b}\,/L \qquad a \Rightarrow a}{a/b, b/c, c \Rightarrow a}\,/L}{a/b, b/c \Rightarrow a/c}\,/R}{a/b \Rightarrow (a/c)/(b/c)}\,/R \qquad\qquad \dfrac{\dfrac{\dfrac{c \Rightarrow c \qquad \dfrac{b \Rightarrow b \quad a \Rightarrow a}{a/b, b \Rightarrow a}\,/L}{a/b, b/c, c \Rightarrow a}\,/L}{a/b, b/c \Rightarrow a/c}\,/R}{a/b \Rightarrow (a/c)/(b/c)}\,/R$$

Geach: uniform head-driven search

Of these two, only the first survives in the **L*** regime.

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{(c)^\star \Rightarrow c}\,\star L}{c \Rightarrow (c)^\star}\,\star R \qquad \dfrac{}{(b)^\star \Rightarrow b}\,\star L}{(b/c)^\star, c \Rightarrow b}\,/L}{b/c, c \Rightarrow (b)^\star}\,\star R \qquad \dfrac{}{(a)^\star \Rightarrow a}\,\star L}{(a/b)^\star, b/c, c \Rightarrow a}\,/L}{\dfrac{a/b, b/c, c \Rightarrow (a)^\star}{a/b, b/c \Rightarrow (a/c)^\star}\,/R}\,\star R}{a/b \Rightarrow ((a/c)/(b/c))^\star}\,/R$$

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{(c)^\star \Rightarrow c}\,\star L}{c \Rightarrow (c)^\star}\,\star R \qquad \dfrac{\text{FAIL}}{a/b, (b)^\star \Rightarrow a}}{a/b, (b/c)^\star, c \Rightarrow a}\,/L}{a/b, b/c, c \Rightarrow (a)^\star}\,\star R}{a/b, b/c \Rightarrow (a/c)^\star}\,/R}{a/b \Rightarrow ((a/c)/(b/c))^\star}\,/R$$

Uniform head-driven search: modal control

We interleave the proof unfolding and the unpacking of the $(\cdot)^1, (\cdot)^0$ translations.

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\text{to be cont'd}}{(a)^1/(b)^0, \Box^{\downarrow}(b/c)^1, \Box^{\downarrow}(c)^1 \Rightarrow a}}{(a/b)^1, \Box^{\downarrow}(b/c)^1, \Box^{\downarrow}(c)^1 \Rightarrow a}\,(\cdot)^1}{(\Box^{\downarrow}(a/b)^1)^{\diamond}, \Box^{\downarrow}(b/c)^1, \Box^{\downarrow}(c)^1 \Rightarrow a}\,\Box^{\downarrow}L}{(\Box^{\downarrow}(a/b)^1, \Box^{\downarrow}(b/c)^1, \Box^{\downarrow}(c)^1)^{\diamond} \Rightarrow a}\,K'}{\Box^{\downarrow}(a/b)^1, \Box^{\downarrow}(b/c)^1, \Box^{\downarrow}(c)^1 \Rightarrow \Box^{\downarrow}a}\,\Box^{\downarrow}R}{\Box^{\downarrow}(a/b)^1, \Box^{\downarrow}(b/c)^1, \Box^{\downarrow}(c)^1 \Rightarrow (a)^0}\,(\cdot)^0}{\Box^{\downarrow}(a/b)^1, \Box^{\downarrow}(b/c)^1 \Rightarrow (a)^0/\Box^{\downarrow}(c)^1}\,/R}{\Box^{\downarrow}(a/b)^1, \Box^{\downarrow}(b/c)^1 \Rightarrow (a/c)^0}\,(\cdot)^0}{\Box^{\downarrow}(a/b)^1 \Rightarrow (a/c)^0/\Box^{\downarrow}(b/c)^1}\,/R}{\Box^{\downarrow}(a/b)^1 \Rightarrow ((a/c)/(b/c))^0}\,(\cdot)^0$$

Consider first the interaction of [/R] rules and selection of the active antecedent type. Antecedent types all have $\Box^{\downarrow}$ as main connective. The $\Box^{\downarrow}$ acts as a *lock*: a $\Box^{\downarrow}A$ formula can only become active when it is *unlocked* by the key $\Diamond$ (or $(\cdot)^{\diamond}$ in structural terms). The key becomes available only when the head of the goal formula is reached: through residuation, $[\Box^{\downarrow}R]$ transmits $\Diamond$ to the antecedent, where it selects a formula via $[K']$.

Transmission of the active formula

There is only *one* key $\Diamond$ by residuation on the $\Box^{\downarrow}$ of the goal formula. As soon as it is used to unlock an antecedent formula, that formula has to remain active and connect to the Axiom sequent.

$$
\begin{prooftree}
\AxiomC{$c \Rightarrow c$}
\RightLabel{$(\cdot)^1$}
\UnaryInfC{$(c)^1 \Rightarrow c$}
\RightLabel{$\Box^{\downarrow}L$}
\UnaryInfC{$(\Box^{\downarrow}(c)^1)^{\diamond} \Rightarrow c$}
\RightLabel{$\Box^{\downarrow}R$}
\UnaryInfC{$\Box^{\downarrow}(c)^1 \Rightarrow \Box^{\downarrow}c$}
\RightLabel{$(\cdot)^0$}
\UnaryInfC{$\Box^{\downarrow}(c)^1 \Rightarrow (c)^0$}
\AxiomC{$b \Rightarrow b$}
\RightLabel{$(\cdot)^1$}
\UnaryInfC{$(b)^1 \Rightarrow b$}
\RightLabel{$/L$}
\BinaryInfC{$(b)^1/(c)^0, \Box^{\downarrow}(c)^1 \Rightarrow b$}
\RightLabel{$(\cdot)^1$}
\UnaryInfC{$(b/c)^1, \Box^{\downarrow}(c)^1 \Rightarrow b$}
\RightLabel{$\Box^{\downarrow}L$}
\UnaryInfC{$(\Box^{\downarrow}(b/c)^1)^{\diamond}, \Box^{\downarrow}(c)^1 \Rightarrow b$}
\RightLabel{K'}
\UnaryInfC{$(\Box^{\downarrow}(b/c)^1, \Box^{\downarrow}(c)^1)^{\diamond} \Rightarrow b$}
\RightLabel{$\Box^{\downarrow}R$}
\UnaryInfC{$\Box^{\downarrow}(b/c)^1, \Box^{\downarrow}(c)^1 \Rightarrow \Box^{\downarrow}b$}
\RightLabel{$(\cdot)^0$}
\UnaryInfC{$\Box^{\downarrow}(b/c)^1, \Box^{\downarrow}(c)^1 \Rightarrow (b)^0$}
\AxiomC{$a \Rightarrow a$}
\RightLabel{$(\cdot)^1$}
\UnaryInfC{$(a)^1 \Rightarrow a$}
\RightLabel{$/L$}
\BinaryInfC{$(a)^1/(b)^0, \Box^{\downarrow}(b/c)^1, \Box^{\downarrow}(c)^1 \Rightarrow a$}
\end{prooftree}
$$

Below, we show how the wrong identification of the antecedent head leads to failure. The key to unlock $\Box^{\downarrow}(a/b)^1$ has been spent on the wrong formula. As a result, the implication in $(a/b)^1$ cannot become active. Compare with the failure of the corresponding **L*** derivation above.

$$
\begin{prooftree}
\AxiomC{$c \Rightarrow c$}
\RightLabel{$(\cdot)^1$}
\UnaryInfC{$(c)^1 \Rightarrow c$}
\RightLabel{$\Box^{\downarrow}L$}
\UnaryInfC{$(\Box^{\downarrow}(c)^1)^{\diamond} \Rightarrow c$}
\RightLabel{$\Box^{\downarrow}R$}
\UnaryInfC{$\Box^{\downarrow}(c)^1 \Rightarrow \Box^{\downarrow}c$}
\RightLabel{$(\cdot)^0$}
\UnaryInfC{$\Box^{\downarrow}(c)^1 \Rightarrow (c)^0$}
\AxiomC{FAILS}
\RightLabel{$\dagger$}
\UnaryInfC{$\Box^{\downarrow}(a/b)^1, (b)^1 \Rightarrow a$}
\RightLabel{$/L$}
\BinaryInfC{$\Box^{\downarrow}(a/b)^1, (b)^1/(c)^0, \Box^{\downarrow}(c)^1 \Rightarrow a$}
\RightLabel{$(\cdot)^1$}
\UnaryInfC{$\Box^{\downarrow}(a/b)^1, (b/c)^1, \Box^{\downarrow}(c)^1 \Rightarrow a$}
\RightLabel{$\Box^{\downarrow}L$}
\UnaryInfC{$\Box^{\downarrow}(a/b)^1, (\Box^{\downarrow}(b/c)^1)^{\diamond}, \Box^{\downarrow}(c)^1 \Rightarrow a$}
\RightLabel{K'}
\UnaryInfC{$(\Box^{\downarrow}(a/b)^1, \Box^{\downarrow}(b/c)^1, \Box^{\downarrow}(c)^1)^{\diamond} \Rightarrow a$}
\RightLabel{$\Box^{\downarrow}R$}
\UnaryInfC{$\Box^{\downarrow}(a/b)^1, \Box^{\downarrow}(b/c)^1, \Box^{\downarrow}(c)^1 \Rightarrow \Box^{\downarrow}a$}
\end{prooftree}
$$

2.2 Mixed inference: multimodal systems

Our second generalizing move is from mixed (2,3) frames (unimodal for each arity) to *multimodal* frames. The objective here is to combine the virtues of the distinct logics we have discussed before in one multimodal system, and at the same time to overcome the limitations of the individual systems in isolation. Each of the component logics has its own specific resource management properties: when combining the different logics, we have to take care that these individual characteristics are left intact. We do this by relativizing linguistic composition to specific resource management *modes*. But also, we want the inferential capacity of the combined logic to be more than the sum of the parts. The extra expressivity comes from *interaction postulates* that hold when different modes are in construction with one another.

We have treated the multimodal style of inference and its linguistic applications in greater detail elsewhere (cf [24, 25, 23]). In the present section we draw special attention to the interaction postulates, highlighting the correspondence between the type (1,2) K distributivity principles discussed in §2.1 and the type (2,2) principles regulating communication among binary modes.

On the syntactic level, the category formulae for the multimodal system are defined inductively on the basis of a set of category atoms $\mathcal{A}$ and a set of indices

I as shown below. We refer to the $i \in I$ as *resource management modes*, or modes for short.

$$\mathcal{F} ::= \mathcal{A} \mid \mathcal{F}/_i\mathcal{F} \mid \mathcal{F} \bullet_i \mathcal{F} \mid \mathcal{F}\backslash_i\mathcal{F} \mid \Diamond_i\mathcal{F} \mid \Box_i^{\downarrow}\mathcal{F}$$

The semantics for the mixed language is a straightforward generalisation of frame semantics for the simple systems. Rather than interpret multiplicative connectives in terms of *one* (binary, ternary) privileged notion of linguistic composition, we throw different forms of linguistic composition together and interpret in multimodal frames $\langle W, \{R_i^2\}_{i\in I}, \{R_i^3\}_{i\in I}\rangle$. A valuation on a frame respects the structure of the complex types in the familiar way, interpreting each of the modes $i \in I$ with its own (binary or ternary) accesibility relation.

$$\begin{array}{lcl}
v(\Diamond_i A) & = & \{x \mid \exists y(R_i xy \ \wedge \ y \in v(A)\} \\
v(\Box_i^{\downarrow} A) & = & \{x \mid \forall y(R_i yx \ \Rightarrow y \in v(A)\} \\
v(A \circ_i B) & = & \{z \mid \exists x \exists y[R_i zxy \ \& \ x \in v(A) \ \& \ y \in v(B)]\} \\
v(C/_i B) & = & \{x \mid \forall y \forall z[(R_i zxy \ \& \ y \in v(B)) \Rightarrow z \in v(C)]\} \\
v(A\backslash_i C) & = & \{y \mid \forall x \forall z[(R_i zxy \ \& \ x \in v(A)) \Rightarrow z \in v(C)]\}
\end{array}$$

We can present the multimodal logic axiomatically or in Gentzen style. In the axiomatic presentation, we have the familiar residuation pattern now relativized to resource management modes:

$$\Diamond_i A \to B \quad \textit{iff} \quad A \to \Box_i^{\downarrow} B$$
$$A \to C/_i B \quad \textit{iff} \quad A \circ_i B \to C \quad \textit{iff} \quad B \to A\backslash_i C$$

In sequent presentation, each residuated family of multiplicatives $\{\Diamond_i, \Box_i^{\downarrow}\}$, $\{/_i, \circ_i, \backslash_i\}$ has a matching structural connective, again relativized to resource management modes. Antecedent terms are inductively defined as

$$\mathcal{T} ::= \mathcal{F} \mid (\mathcal{T}, \mathcal{T})^i \mid (\mathcal{T})^i$$

Logical rules insist that use and proof of connectives respect the resource management modes.

Multimodal sequent calculus

$$[\mathrm{Ax}]\frac{}{A \Rightarrow A} \qquad \frac{\Gamma \Rightarrow A \quad \Delta[A] \Rightarrow C}{\Delta[\Gamma] \Rightarrow C}[\mathrm{Cut}]$$

$$[\mathrm{R}\Diamond_i]\frac{\Gamma \Rightarrow A}{(\Gamma)^i \Rightarrow \Diamond_i A} \qquad \frac{\Gamma[(A)^i] \Rightarrow B}{\Gamma[\Diamond_i A] \Rightarrow B}[\mathrm{L}\Diamond_i]$$

$$[\mathrm{R}\Box_i^{\downarrow}]\frac{(\Gamma)^i \Rightarrow A}{\Gamma \Rightarrow \Box_i^{\downarrow} A} \qquad \frac{\Gamma[A] \Rightarrow B}{\Gamma[(\Box_i^{\downarrow} A)^i] \Rightarrow B}[\mathrm{L}\Box_i^{\downarrow}]$$

$$[\mathrm{R}/_i]\frac{(\Gamma, B)^i \Rightarrow A}{\Gamma \Rightarrow A/_i B} \qquad \frac{\Gamma \Rightarrow B \qquad \Delta[A] \Rightarrow C}{\Delta[(A/_i B, \Gamma)^i] \Rightarrow C}[\mathrm{L}/_i]$$

$$[\mathrm{R}\backslash_i]\frac{(B, \Gamma)^i \Rightarrow A}{\Gamma \Rightarrow B\backslash_i A} \qquad \frac{\Gamma \Rightarrow B \qquad \Delta[A] \Rightarrow C}{\Delta[(\Gamma, B\backslash_i A)^i] \Rightarrow C}[\mathrm{L}\backslash_i]$$

$$[\mathrm{L}\circ_i]\frac{\Gamma[(A, B)^i] \Rightarrow C}{\Gamma[A \circ_i B] \Rightarrow C} \qquad \frac{\Gamma \Rightarrow A \qquad \Delta \Rightarrow B}{(\Gamma, \Delta)^i \Rightarrow A \circ_i B}[\mathrm{R}\circ_i]$$

In addition to the residuation inferences which are shared by all resource management modes, there are mode-specific structural options. In axiomatic style, they take the form of structural axioms; in sequent presentation, we have the corresponding structural rules. As an illustration, see the structural axioms/rules for a commutative mode c. In the semantics the R_c interpreting this connective will be constrained to satisfy $(\forall x, y, z \in W) R_c zxy \Rightarrow R_c zyx$.

$$A \circ_c B \longleftrightarrow B \circ_c A \quad \frac{\Gamma[(\Delta_2, \Delta_1)^c] \Rightarrow A}{\Gamma[(\Delta_1, \Delta_2)^c] \Rightarrow A}[\text{Comm}]$$

Multimodal communication principles

What we have done so far is simply put together the individual systems discussed before in isolation. This is enough to gain combined access to the inferential capacities of the component logics, and one avoids the unpleasant collapse into the least discriminating logic that results from combining logics without taking into account the mode specifications, cf our discussion of CCG in §1. But as things are, the borders between the constituting logics in our multimodal setting are still hermetically closed. Let us turn then to the question of multimodal communication.

Communication between different modes is obtained semantically by frame conditions linking R_i and R_j. Two types of constraints can be distinguished:

- ordering of the accesibility relations R_i in terms of the information they provide about the structure of the linguistic resources. In terms of frame conditions, say for R_i^3 more informative than R_j^3, this means $(\forall x, y, z \in W) R_i xyz \Rightarrow R_j xyz$. For example, since the non-commutative product $\bullet$ is more informative than the commutative product $\otimes$, we'd have the constraint: $(\forall x, y, z) R_\bullet xyz \Rightarrow R_\otimes xyz$. Corresponding to this form of frame condition, there will be Inclusion Postulates in the logic.

- frame conditions 'mixing' distinct R_i implementing, for example, mixed associativity or commutativity laws. This form of constraint is expressed in the logic in terms of Interaction Postulates.

We discuss them in turn.

Inclusion: products

As a simple example of an inclusion constraint, consider the following cut-free Gentzen derivation showing that commutative $\otimes$ is derivable from non-commutative $\bullet$ product. The derivation uses Left and Right logical rules (which depend on the proper sequent punctuation), the inclusion structural rule connecting $\bullet$-type and $\otimes$-type configurations, and the Permutation structural rule for $\otimes$-type configurations.

$$\cfrac{\cfrac{\cfrac{\cfrac{B \Rightarrow B \quad A \Rightarrow A}{(B, A)^{\otimes} \Rightarrow B \otimes A}\,\otimes R}{(A, B)^{\otimes} \Rightarrow B \otimes A}\,P\otimes}{(A, B)^{\bullet} \Rightarrow B \otimes A}\,\bullet \sqsubseteq \otimes}{A \bullet B \Rightarrow B \otimes A}\,\bullet L$$

Compare the combinatory derivation, which depends on non-logical axiom schemata ($\pi_\otimes$ and the inclusion postulate $\lambda_{\bullet,\otimes}$) and transitivity.

$$\frac{\lambda_{\bullet,\otimes} : A \bullet B \to A \otimes B \quad \pi_\otimes : A \otimes B \to B \otimes A}{\pi_\otimes \circ \lambda_{\bullet,\otimes} : A \bullet B \to B \otimes A}$$

Inclusion: order reversal through residuation

Where the inclusion for products gives $A \bullet_i B \rightarrow A \bullet_j B$, derivability for the residual implications is the other way around. Below the combinator derivation and its cut-free Gentzen variant.

$$\dfrac{\lambda_{i,j} : A/_jB \bullet_i B \rightarrow A/_jB \bullet_j B \quad \dfrac{\mathbf{id} : A/_jB \rightarrow A/_jB}{\beta_j^{-1}(\mathbf{id}) : A/_jB \bullet_j B \rightarrow A}}{\dfrac{\beta_j^{-1}(\mathbf{id}) \circ \lambda_{i,j} : A/_jB \bullet_i B \rightarrow A}{\beta_i(\beta_j^{-1}(\mathbf{id}) \circ \lambda_{i,j}) : A/_jB \rightarrow A/_iB}}$$

$$\dfrac{\dfrac{\dfrac{B \Rightarrow B \quad A \Rightarrow A}{(A/_jB, B)^j \Rightarrow A}\ /_jL}{(A/_jB, B)^i \Rightarrow A}\ \bullet_i \sqsubseteq \bullet_j}{A/_jB \Rightarrow A/_iB}\ /_iR$$

To convince oneself that the intuitions are satisfied here, let mode i be non-commutative, and mode j commutative, as in our previous example. Now consider a conjunction of commutative $A/_jB$ and non-commutative $A/_iB$. Given the direction of the derivability arrow established above, the conjunction as a whole is $A/_iB$ — which is as it should be: the commutativity of the j mode should not spoil the more discriminating resource management of the i mode.

The reader may be interested in comparing the treatment of inclusion constraints given here with that of Hepple [17] where the derivability arrows are systematically reversed.

Interaction principles: type (1,2)

In the discussion of (2,3) frames in §2.1 we have already seen the K principles of (strong, weak) distributivity for the communication between unary $\Diamond, \Box^\downarrow$ and binary $/, \bullet, \backslash$. In the multimodal setting, these principles can be relativized to specific mode interactions, resulting in a vastly increased control over resource management.

$$\begin{array}{rl} K1_{i,j}: & \Diamond_i(A \bullet_j B) \rightarrow \Diamond_i A \bullet_j B \\ K2_{i,j}: & \Diamond_i(A \bullet_j B) \rightarrow A \bullet_j \Diamond_i B \\ K_{i,j}: & \Diamond_i(A \bullet_j B) \rightarrow \Diamond_i A \bullet_j \Diamond_i B \end{array}$$

Below the correponding frame conditions ($\forall x, y, z, w \in W$).

$$\begin{array}{rl} K(1,2)_{i,j}: & (R_iwx \ \&\ R_jxyz) \Rightarrow \exists y'(R_iy'y \ \&\ R_jwy'z) \ \vee\ \exists z'(R_iz'z \ \&\ R_jwyz') \\ K_{i,j}: & (R_iwx \ \&\ R_jxyz) \Rightarrow \exists y'\exists z'(R_iy'y \ \&\ R_iz'z \ \&\ R_jwy'z') \end{array}$$

A linguistic application of relativized K principles can be found in [23], where a head-selection $\Diamond$ interacts via $K1$ with left-headed dependency product $\bullet_l$, and via $K2$ with right-headed dependency product $\bullet_r$.

Interaction principles: type (2,2)

Consider finally interaction principles for inter-mode communication among binary families. Above, we distinguished weak and strong distributivity principles for (1,2) type interactions. Let us do exactly the same for interaction of the (2,2) type.

Consider first interaction of the weak distributivity type. Below one finds principles of *mixed* associativity and commutativity. Instead of the global associativity and commutativity options characterizing **L**, **NLP**, **LP**, we find mixed versions constrained to the situation where modes i and j are in construction. (Symmetric duals can be added, with the j mode adjunction distributing from the left.)

$$\begin{array}{lll} MC: & (A \bullet_i B) \bullet_j C \longleftrightarrow (A \bullet_j C) \bullet_i B & : MC' \\ MA: & (A \bullet_i B) \bullet_j C \longleftrightarrow A \bullet_i (B \bullet_j C) & : MA' \end{array}$$

The postulates correspond to frame conditions ($\forall xyzuv \in W$):

$$\begin{array}{rl} MC: & (R_i xyz \ \& \ R_j zuv) \Rightarrow \exists t(R_j xut \ \& \ R_i tyv) \\ MA: & (R_i xyz \ \& \ R_j zuv) \Rightarrow \exists t(R_j yut \ \& \ R_i xtv) \\ MC': & (R_j xyz \ \& \ R_i zuv) \Rightarrow \exists t(R_i xut \ \& \ R_j tyv) \\ MA': & (R_j xyz \ \& \ R_i uzv) \Rightarrow \exists t(R_i uxt \ \& \ R_j tyv) \end{array}$$

And they manifest themselves in structural rules in Gentzen presentation.

$$\frac{\Gamma[((\Delta_1,\Delta_3)^j,\Delta_2)^i] \Rightarrow A}{\Gamma[((\Delta_1,\Delta_2)^i,\Delta_3)^j] \Rightarrow A}\ \mathrm{MC} \qquad \frac{\Gamma[(\Delta_1,(\Delta_2,\Delta_3)^j)^i] \Rightarrow A}{\Gamma[((\Delta_1,\Delta_2)^i,\Delta_3)^j] \Rightarrow A}\ \mathrm{MA}$$

$$\frac{\Gamma[((\Delta_1,\Delta_2)^i,\Delta_3)^j] \Rightarrow A}{\Gamma[((\Delta_1,\Delta_3)^j,\Delta_2)^i] \Rightarrow A}\ \mathrm{MC'} \qquad \frac{\Gamma[((\Delta_1,\Delta_2)^i,\Delta_3)^j] \Rightarrow A}{\Gamma[(\Delta_1,(\Delta_2,\Delta_3)^j)^i] \Rightarrow A}\ \mathrm{MA'}$$

Application: mixed composition

For linguistic application of these general postulates, we refer to [25, 23], where it is shown that a multimodal variant of CCG 'mixed composition' laws — which in the absence of mode constraints cause collapse of **L** into **LP**, as we saw above — are theorems in combined logics with the MC/MA principles. Schematically, in 'Geach' version, we have the following valid type transition (and the symmetric case), where the w mode stands for head adjunction and the r mode for righ-headed dependency adjunction.

$$A/_w B \Rightarrow (C\backslash_r A)/_w (C\backslash_r B)$$

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{C \Rightarrow C \quad B \Rightarrow B}{(C, C\backslash_r B)^r \Rightarrow B}\ \backslash_r L \qquad A \Rightarrow A}{(A/_w B, (C, C\backslash_r B)^r)^w \Rightarrow A}\ /_w L}{(C, (A/_w B, C\backslash_r B)^w)^r \Rightarrow A}\ MC}{(A/_w B, C\backslash_r B)^w \Rightarrow C\backslash_r A}\ \backslash_r R}{A/_w B \Rightarrow (C\backslash_r A)/_w (C\backslash_r B)}\ /_w R}{}$$

Type (2,2) interaction: strong distributivity

The weak distributivity principles MC, MA keep us within the family of resource neutral logics: they do not affect the multiplicity of the resources in a configuration. Strong distributivity principles of the (2,2) type are not resource neutral: they duplicate resources. But in the multimodal setting, this gives access to mode-controlled forms of Contraction.

We give one illustration. Consider the distributivity principle S below, which strongly distributes mode j over mode i thus copying a C datum.

$$S: \quad (A \bullet_i B) \bullet_j C \to (A \bullet_j C) \bullet_i (B \bullet_j C)$$

We leave it to the reader to compute frame conditions and structural rules. But we close this section pointing out that the **S** combinator of CCG (used in the derivation of parasitic gaps) is an immediate consequence of the S postulate. Again, in a unimodal setting the **S** combinator in combination with Residuation causes disaster. (Abbreviation: $D = ((A/_iB)/_jC \bullet_j C) \bullet_i ((B/_jC) \bullet_j C)$.)

$$\dfrac{\dfrac{S : ((A/_iB)/_jC \bullet_i B/_jC) \bullet_j C \to D \quad f : D \to A}{f \circ S : ((A/_iB)/_jC \bullet_i B/_jC) \bullet_j C \to A}}{\beta_j(f \circ S) : (A/_iB)/_jC \bullet_i B/_jC \to A/_jC}$$

3 Conclusion

This paper is a technical investigation of the architecture of mixed categorial type logics. The *raison d'être* for such an exercise, we hope to have made clear, is the linguistic application of these logics to problems of grammatical analysis — without the linguistic motivation many of the logical issues addressed above would simply not arise. At the same time, we would like to stress that progress on the descriptive level of linguistic analysis is really dependent on a clear understanding of the logical structure of the type systems employed.

A Appendix: Invariants

Type transformations $A \to B$ in the Lambek systems **NL**, **L**, **NLP**, **LP** preserve the resource sensitivity properties of the individual logics. As a result one can characterize necessary conditions for derivability $A \to B$ in the different Lambek logics in terms of a number of useful invariants. In this Appendix, we present the known invariants for **L** and **LP** and add new ones for the systems **NL** and **NLP**, and for the extension with unary $\Diamond, \Box^\downarrow$ connectives.

A.1 Preservation properties: NL, L, NLP, LP

On the most general level, all members of the family of occurrence logics share the property of *resource* preservation: in a derivation $A \to B$ there is an equilibrium between available and consumed resources, i.e. between positive and negative occurrences of atomic type formulae. The atom count of Van Benthem [2] measures this property. We present the stronger invariants in a format due to Pentus [29] which translates type formulae into elements of the free group generated by the type atoms. First, resource preservation can be sharpened to *order preservation* for the systems **L** and **NL** where derivability is dependent on the linear order of the resources (Roorda, [30]). Order preservation, in its turn, can be sharpened to *structure preservation* for **NL**, where apart from linear order also the hierarchical grouping of resources determines derivability. In the setting of Abelian groups, finally, we obtain the **NLP** version of structure preservation, and an alternative way of expressing resource preservation.

Notice carefully that the invariants to be discussed here provide necessary but not sufficient conditions for derivability: they do not take into account the difference between $A \to B$ and $B \to A$. They will be typically used then in arguments about *non*-derivability, and in a more practical context, in pruning the search space for categorial theorem proving, where the invariants allow early detection of non-theorems.

Proposition A.1 *Resource preservation. (Atom count, Van Benthem.) Let $\#_p(A)$ be a function counting positive and negative occurrences of atomic formulae p in arbitrary formulae A defined as below.*

$$\#_p(p) = 1, \quad \#_p(q) = 0 \quad \textit{for atoms } p, q,\ p \neq q$$
$$\#_p(A/B) = \#_p(B\backslash A) = \#_p(A) - \#_p(B), \quad \#_p(A \bullet B) = \#_p(A) + \#_p(B)$$

$$\mathbf{NL},\ \mathbf{L},\ \mathbf{NLP},\ \mathbf{LP} \vdash A \rightarrow B \text{ implies } \#_p(A) = \#_p(B) \text{ for any } p \in \mathcal{A}.$$

The proof of Prop A.1, and of the invariants to be discussed below, is obtained by straightforward induction on derivations using the axiomatic presentation of the logics. The axioms can be shown by inspection to have the relevant property (the logical axiom $A \rightarrow A$ plus structural postulates, if any), and the rules of inference (transitivity, residuation) preserve it.

For the order preservation invariant, we interpret type formulae as elements of $\mathcal{G}(\mathcal{A})$, the free group generated by the type atoms $\mathcal{A}$. The group is equiped with an associative operation '$\cdot$' and with a two-sided identity element e satisfying $e \cdot p = p = p \cdot e$. As an element of the free group each prime formula p has an inverse p^{-1} such that $p^{-1} \cdot p = e = p \cdot p^{-1}$. The translations $(\cdot)^1, (\cdot)^0$ are each other's opposite in the sense that $(A)^1(A)^0 = e = (A)^0(A)^1$ for arbitrary types A.

Proposition A.2 *Order preservation ('Balance', Roorda, Pentus.) Let $\mathcal{G}(\mathcal{A})$ be the free group generated by the atomic type formulae $\mathcal{A}$. Define translations $(\cdot)^1, (\cdot)^0 : \mathcal{F} \mapsto \mathcal{G}(\mathcal{A})$:*

$$\begin{array}{rl} (p)^1 = p, & (p)^0 = p^{-1} \\ (A \bullet B)^1 = (A)^1 \cdot (B)^1, & (A \bullet B)^0 = (B)^0 \cdot (A)^0 \\ (A/B)^1 = (A)^1 \cdot (B)^0, & (A/B)^0 = (B)^1 \cdot (A)^0 \\ (B\backslash A)^1 = (B)^0 \cdot (A)^1, & (B\backslash A)^0 = (A)^0 \cdot (B)^1 \end{array}$$

$$\mathbf{NL},\ \mathbf{L} \vdash A \rightarrow B \text{ implies } (A)^1 \cdot (B)^0 = e.$$

Observe that an alternative version of resource preservation in the sense of Proposition A.1 can be obtained via the free group interpretation of Proposition A.2 by turning $\mathcal{G}(\mathcal{A})$ into an Abelian group, where the group operation '$\cdot$' is commutative as well as associative.

For the system **NL** we need an invariant capturing the fact that derivability is dependent not only on the linear order of the resources but also on their hierarchical structuring. Structure preservation is obtained by doubling up the recursion of the order preservation invariant: instead of mappings $(\cdot)^1$, $(\cdot)^0$, with $(A)^1(A)^0 = e$, we have *four* translation mappings $(\cdot)^1, (\cdot)^0, {}^1(\cdot), {}^0(\cdot)$ set up in such a way that ${}^1(A)(A)^0 = (A)^{0\,1}(A) = {}^0(A)(A)^1 = (A)^{1\,0}(A) = e$ for arbitrary A. The recursion keeps track of whether a subformula is within an even or odd number of brackets. For the basis of the recursion, i.e. the translation of prime formulae, we have a choice. Either (†) we augment the free group interpretation with a pair of elements $\mathbf{1}$, $\mathbf{0}$, to be thought of as structure markers. They are each other's inverses, i.e. $\mathbf{1} \cdot \mathbf{0} = e = \mathbf{0} \cdot \mathbf{1}$. Or (‡) we have for each prime formula p an even (${}^\top p$) and an odd ($p^\top$) representative in the group, together with their inverses ($p^\perp$ and ${}^\perp p$ respectively).

Proposition A.3 *Structure preservation. Let $\mathcal{G}(\mathcal{A} \cup \{\mathbf{1}, \mathbf{0}\})$ be the free group generated by the union of atomic type formulae $\mathcal{A}$ and structure markers $\mathbf{1}, \mathbf{0}$ satisfying $\mathbf{1} \cdot \mathbf{0} = e = \mathbf{0} \cdot \mathbf{1}$ (†). Let $\mathcal{G}(\mathcal{A}^{\perp,\top})$ be the free group generated by the union of even/odd atomic type formula tokens $\mathcal{A}$ satisfying ${}^\top p \cdot p^\perp = e = p^\top \cdot {}^\perp p$ (‡). Define translations $(\cdot)^1, (\cdot)^0, {}^1(\cdot), {}^0(\cdot) : \mathcal{F} \mapsto \mathcal{G}(\mathcal{A} \cup \{\mathbf{1}, \mathbf{0}\})$(†) or $\mathcal{F} \mapsto \mathcal{G}(\mathcal{A}^{\perp,\top})$(‡) as* *follows:*

$$
\begin{array}{lllll}
(\dagger) & (p)^1 = p \cdot \mathbf{1}, & (p)^0 = p^{-1} \cdot \mathbf{0}, & {}^1(p) = \mathbf{1} \cdot p, & {}^0(p) = \mathbf{0} \cdot p^{-1}, \\
(\ddagger) & {}^1(p) = {}^{\top}p, & (p)^0 = p^{\perp}, & (p)^1 = p^{\top}, & {}^0(p) = {}^{\perp}p,
\end{array}
$$

$$
\begin{array}{rl}
{}^1(A \bullet B) = (A)^1 \cdot (B)^1, & (A \bullet B)^0 = {}^0(B) \cdot {}^0(A) \\
(A \bullet B)^1 = {}^1(A) \cdot {}^1(B), & {}^0(A \bullet B) = (B)^0 \cdot (A)^0 \\
{}^1(A/B) = (A)^1 \cdot (B)^0, & (A/B)^0 = {}^1(B) \cdot {}^0(A) \\
(A/B)^1 = {}^1(A) \cdot {}^0(B), & {}^0(A/B) = (B)^1 \cdot (A)^0 \\
{}^1(B\backslash A) = (B)^0 \cdot (A)^1, & (B\backslash A)^0 = {}^0(A) \cdot {}^1(B) \\
(B\backslash A)^1 = {}^0(B) \cdot {}^1(A), & {}^0(B\backslash A) = (A)^0 \cdot (B)^1
\end{array}
$$

$$\mathbf{NL} \vdash A \to B \textit{ implies } {}^1(A) \cdot (B)^0 = e.$$

Example A.4 We check an **NL** theorem, Lifting $p \to q/(p\backslash q)$ and a non-theorem of **NL**, the Geach transformation $p/q \to (p/r)/(q/r)$. Both are order-preserving ($\star$). Structure preservation ($\star\star$) holds for Lifting, not for Geach.

$$
\begin{array}{c}
\mathbf{NL} \vdash \quad p \to q/(p\backslash q) \\
(\star) \quad (p)^1(q/(p\backslash q))^0 = p(p\backslash q)^1(q)^0 = p(p)^0(q)^1q^{-1} = pp^{-1}qq{-}1 = e \\
(\star\star\ \dagger) \quad {}^1(p)(q/(p\backslash q))^0 = \mathbf{1}p^1(p\backslash q)^0(q) = \mathbf{1}p(p)^0(q)^1\mathbf{0}q^{-1} = \mathbf{1}pp^{-1}\mathbf{0}q\mathbf{10}q^{-1} = e
\end{array}
$$

$$
\begin{array}{c}
\mathbf{NL} \not\vdash \quad p/q \to (p/r)/(q/r) \\
(\star) \qquad (p/q)^1((p/r)/(q/r))^0 = \\
(p)^1(q)^0(q/r)^1(p/r)^0 = pq^{-1}(q)^1(r)^0(r)^1(p)^0 = pq^{-1}qr^{-1}rp^{-1} = e \\
(\star\star\ \ddagger) \qquad {}^1(p/q)((p/r)/(q/r))^0 = \\
(p)^1(q)^{0\,1}(q/r)^0(p/r) = p^{\top}q^{\perp}(q)^1(r)^0(r)^1(p)^0 = p^{\top}q^{\perp}q^{\top}r^{\perp}r^{\top}p^{\perp} \neq e
\end{array}
$$

The two versions of the structure preservation invariant each have their merit. On the basis of ($\ddagger$) one obtains the **NLP** variant of structure preservation by interpreting $\mathcal{G}(\mathcal{A}^{\perp,\top})$ as an Abelian group (Kurtonina p.c.). For the ($\dagger$) version, Abelianization does not have the desired effect because the positional information of the structure markers is destroyed by the commutativity of the group operation. But on the basis of the ($\dagger$) version, we can obtain a more economical translation, where structure markers **0**, **1** are cashed out not at the level of prime formulae, but at the points where the recursion breaks up compound formulae. The version of A.5 subsumes a sharpened form of Janssen's ([18]) slash balance, which checks the balance of positive/negative occurrences of implications, ignoring directionality. Prop A.5 generalizes the operator count to the full set of connectives and keeps track of the difference between right and left implications.

Proposition A.5 *Structure preservation with operator balance. Let* $\mathcal{G}(\mathcal{A} \cup \{\mathbf{1}, \mathbf{0}\})$ *be as in Prop A.3 Define translations* $(\cdot)^1, (\cdot)^0, {}^1(\cdot), {}^0(\cdot) : \mathcal{F} \mapsto \mathcal{G}(\mathcal{A} \cup \{\mathbf{1}, \mathbf{0}\})$*:*

$$
\begin{array}{rl}
(p)^1 = p = {}^1(p), & {}^0(p) = p^{-1} = (p)^0 \\
{}^1(A \bullet B) = {}^1(A) \cdot {}^1(B) \cdot \mathbf{1}, & (A \bullet B)^0 = \mathbf{0} \cdot (B)^0 \cdot (A)^0 \\
(A \bullet B)^1 = \mathbf{1} \cdot (A)^1 \cdot (B)^1, & {}^0(A \bullet B) = {}^0(B) \cdot {}^0(A) \cdot \mathbf{0} \\
{}^1(A/B) = {}^1(A) \cdot \mathbf{0} \cdot (B)^0, & (A/B)^0 = {}^1(B) \cdot \mathbf{1} \cdot (A)^0 \\
(A/B)^1 = \mathbf{0} \cdot (A)^1 \cdot {}^0(B), & {}^0(A/B) = (B)^1 \cdot {}^0(A) \cdot \mathbf{1} \\
{}^1(B\backslash A) = (B)^0 \cdot {}^1(A) \cdot \mathbf{0}, & (B\backslash A)^0 = \mathbf{1} \cdot (A)^0 \cdot {}^1(B) \\
(B\backslash A)^1 = {}^0(B) \cdot \mathbf{0} \cdot (A)^1, & {}^0(B\backslash A) = {}^0(A) \cdot \mathbf{1} \cdot (B)^1
\end{array}
$$

$$\mathbf{NL} \vdash A \to B \textit{ implies } {}^1(A) \cdot (B)^0 = e.$$

Example A.6 We compare the first ($\dagger$) and second ($\ddagger$) versions of structure preservation for an **NL** theorem, Lifting $p \to q/(p\backslash q)$ and a non-theorem of **NL**, the Geach transformation $p/q \to (p/r)/(q/r)$.

$$\mathbf{NL} \vdash \quad p \to q/(p\backslash q)$$
$$(\dagger) \quad {}^1(p)(q/(p\backslash q))^0 = \mathbf{1}p^1(p\backslash q)^0(q) = \mathbf{1}p(p)^0(q)^1\mathbf{0}q^{-1} = \mathbf{1}pp^{-1}\mathbf{0}q\mathbf{1}\mathbf{0}q^{-1} = e$$
$$(\ddagger) \quad {}^1(p)(q/(p\backslash q))^0 = p^1(p\backslash q)\mathbf{1}(q)^0 = p(p)^{01}(q)\mathbf{0}\mathbf{1}q^{-1} = pp^{-1}q\mathbf{0}\mathbf{1}q^{-1} = e$$

$$\mathbf{NL} \nvdash \quad p/q \to (p/r)/(q/r)$$
$$(\dagger) \qquad {}^1(p/q)((p/r)/(q/r))^0 =$$
$$(p)^1(q)^{01}(q/r)^0(p/r) = p\mathbf{1}q^{-1}\mathbf{0}(q)^1(r)^0(r)^1(p)^0 = p\mathbf{1}q^{-1}\mathbf{0}q\mathbf{1}r^{-1}\mathbf{0}r\mathbf{1}p^{-1}\mathbf{0} \neq e$$
$$(\ddagger) \qquad {}^1(p/q)((p/r)/(q/r))^0 =$$
$${}^1(p)\mathbf{0}(q)^{01}(q/r)\mathbf{1}(p/r)^0 = p\mathbf{0}q^{-11}(q)\mathbf{0}(r)^0\mathbf{1}^1(r)\mathbf{1}(p)^0 = p\mathbf{0}q^{-1}q\mathbf{0}r^{-1}\mathbf{1}r\mathbf{1}p^{-1} \neq e$$

The table below summarizes the situation of invariants for the Lambek logics in terms of the free group interpretation.

LOGIC	PRESERVATION	PROPOSITION	GROUP OPERATION
LP, L, NLP, NL	Resources	Prop A.2	abelian
L, NL	Order	Prop A.2	non-abelian
NL, NLP	Structure	Prop A.3 (†)	abelian
NL	Structure, Order	Prop A.3 (‡), A.5	non-abelian

Table 1: Invariants

Illustrations

The reader may find it helpful to perform the calculations of the free group interpretation in terms of formula decomposition trees. Below we give some examples, starting with failure of structure preservation for Associativity (product variant and Geach variant).

$$\dfrac{\dfrac{a}{{}^1(a)} \quad \dfrac{\dfrac{b}{{}^1(b)} \quad \dfrac{c}{{}^1(c)} \quad 1}{{}^1(b \bullet c)} \quad 1}{{}^1(a \bullet (b \bullet c))} \qquad \dfrac{0 \quad \dfrac{c^{-1}}{(c)^0} \quad \dfrac{0 \quad \dfrac{b^{-1}}{(b)^0} \quad \dfrac{a^{-1}}{(a)^0}}{(a \bullet b)^0}}{((a \bullet b) \bullet c)^0}$$

$$\mathbf{NL} \nvdash a \bullet (b \bullet c) \to (a \bullet b) \bullet c$$

$$a \cdot b \cdot c \cdot 1 \cdot 1 \cdot 0 \cdot c^{-1} \cdot 0 \cdot b^{-1} \cdot a^{-1} \neq e$$

$$\dfrac{\dfrac{a}{{}^1(a)} \quad 0 \quad \dfrac{b^{-1}}{(b)^0}}{{}^1(a/b)} \qquad \dfrac{\dfrac{\dfrac{b}{{}^1(b)} \quad 0 \quad \dfrac{c^{-1}}{(c)^0}}{{}^1(b/c)} \quad 1 \quad \dfrac{\dfrac{c}{{}^1(c)} \quad 1 \quad \dfrac{a^{-1}}{(a)^0}}{(a/c)^0}}{((a/c)/(b/c))^0}$$

$$\mathbf{NL} \nvdash a/b \to (a/c)/(b/c)$$

$$a \cdot 0 \cdot b^{-1} \cdot b \cdot 0 \cdot c^{-1} \cdot 1 \cdot c \cdot 1 \cdot a^{-1} \neq e$$

Structure preserving commutativity in **NLP**.

$$\dfrac{\dfrac{a^{\top}}{(a)^1} \quad \dfrac{\dfrac{{}^{\top}b}{{}^1(b)} \quad \dfrac{{}^{\top}c}{{}^1(c)}}{(b \bullet c)^1}}{{}^1(a \bullet (b \bullet c))} \qquad \dfrac{\dfrac{{}^{\perp}a}{{}^0(a)} \quad \dfrac{\dfrac{b^{\perp}}{(b)^0} \quad \dfrac{c^{\perp}}{(c)^0}}{{}^0(c \bullet b)}}{((c \bullet b) \bullet a)^0}$$

$$\mathbf{NLP} \vdash a \bullet (b \bullet c) \to (c \bullet b) \bullet a$$

$$a^{\top} \cdot {}^{\top}b \cdot {}^{\top}c \cdot {}^{\perp}a \cdot b^{\perp} \cdot c^{\perp} = a^{\top} \cdot {}^{\perp}a \cdot b^{\perp} \cdot {}^{\top}b \cdot c^{\perp} \cdot {}^{\top}c = e$$

Violation of structure preservation in **NLP**.

$$\dfrac{\dfrac{a^{\top}}{(a)^1} \quad \dfrac{\dfrac{{}^{\top}b}{{}^1(b)} \quad \dfrac{{}^{\top}c}{{}^1(c)}}{(b \bullet c)^1}}{{}^1(a \bullet (b \bullet c))} \qquad \dfrac{\dfrac{\dfrac{b^{\perp}}{(b)^0} \quad \dfrac{a^{\perp}}{(a)^0}}{{}^0(a \bullet b)} \quad \dfrac{{}^{\perp}c}{{}^0(c)}}{(c \bullet (a \bullet b))^0}$$

$$\mathbf{NLP} \not\vdash a \bullet (b \bullet c) \to c \bullet (a \bullet b)$$

$$a^{\top} \cdot {}^{\top}b \cdot {}^{\top}c \cdot b^{\perp} \cdot a^{\perp} \cdot {}^{\perp}c \neq e$$

A.2 Invariants: unary connectives

How do the invariants discussed above extend to the language $\mathcal{F}(/, \bullet, \backslash, \Diamond, \Box^{\downarrow})$?. For the combination of the pure logic of residuation for $\Diamond, \Box^{\downarrow}$ (no frame conditions on R^2) with **NL** (no frame conditions on R^3), and with **L** (Associativity for R^3) we have the following result, which is significant in the light of the modal embedding of **NL** in **L**$\Diamond$ discussed in the main text. Other combinations are left as a topic for further research.

Let $\mathcal{G}(\mathcal{A} \cup \{\mathbf{1}, \mathbf{0}\})$ (†) and $\mathcal{G}(\mathcal{A}^{\top,\perp} \cup \{\mathbf{1}, \mathbf{0}\})$ (‡) have the group theoretic interpretation discussed in the previous section. Define translations $(\cdot)^1, (\cdot)^0, {}^1(\cdot), {}^0(\cdot) : \mathcal{F} \mapsto \mathcal{G}(\mathcal{A} \cup \{\mathbf{1}, \mathbf{0}\})$ (†) or $\mathcal{F} \mapsto \mathcal{G}(\mathcal{A}^{\top,\perp} \cup \{\mathbf{1}, \mathbf{0}\})$ (‡) for the extended modal language as follows:

$$\begin{array}{rl} (\dagger) & (p)^1 = p = {}^1(p), \qquad {}^0(p) = p^{-1} = (p)^0 \\ (\ddagger) & {}^1(p) = {}^{\top}p, \quad (p)^0 = p^{\perp}, \qquad (p)^1 = p^{\top}, \quad {}^0(p) = {}^{\perp}p \end{array}$$

$$\begin{array}{rl} {}^1(\Diamond A) = (A)^1 \cdot \mathbf{1}, & (\Diamond A)^0 = \mathbf{0} \cdot {}^0(A) \\ (\Diamond A)^1 = \mathbf{1} \cdot {}^1(A), & {}^0(\Diamond A) = (A)^0 \cdot \mathbf{0} \\ {}^1(\Box^{\downarrow} A) = \mathbf{0} \cdot (A)^1, & (\Box^{\downarrow} A)^0 = {}^0(A) \cdot \mathbf{1} \\ (\Box^{\downarrow} A)^1 = {}^1(A) \cdot \mathbf{0}, & {}^0(\Box^{\downarrow} A) = \mathbf{1} \cdot (A)^0 \end{array}$$

$$\begin{array}{rl} {}^1(A \bullet B) = (A)^1 \cdot (B)^1, & (A \bullet B)^0 = {}^0(B) \cdot {}^0(A) \\ (A \bullet B)^1 = {}^1(A) \cdot {}^1(B), & {}^0(A \bullet B) = (B)^0 \cdot (A)^0 \\ {}^1(A/B) = (A)^1 \cdot (B)^0, & (A/B)^0 = {}^1(B) \cdot {}^0(A) \\ (A/B)^1 = {}^1(A) \cdot {}^0(B), & {}^0(A/B) = (B)^1 \cdot (A)^0 \\ {}^1(B\backslash A) = (B)^0 \cdot (A)^1, & (B\backslash A)^0 = {}^0(A) \cdot {}^1(B) \\ (B\backslash A)^1 = {}^0(B) \cdot {}^1(A), & {}^0(B\backslash A) = (A)^0 \cdot (B)^1 \end{array}$$

$$\mathbf{L}\Diamond \vdash A \to B \text{ implies } {}^1(A) \cdot (B)^0 = e.$$

(We present the **NL** and **L** case here in uniform fashion. If one is just interested in the **L** case, one can take the simpler Roorda marking, augmented with the **1,0** clauses for the $\Diamond, \Box^{\downarrow}$ formulae.)

In the following illustrations we show how the structure-preservation translation of Prop A.5 for **NL** theorems $A \to B$ and the embedding translation presented above for $(A)^{\sharp} \to (B)^{\sharp}$ in **L**$\Diamond$ have the same group-theoretic interpretation.

Failure of associativity: NL

$$\dfrac{\dfrac{a}{{}^1(a)} \quad \dfrac{\dfrac{b}{{}^1(b)} \quad \dfrac{c}{{}^1(c)} \quad 1}{{}^1(b \bullet c)} \quad 1}{{}^1(a \bullet (b \bullet c))} \qquad \dfrac{0 \quad \dfrac{c^{-1}}{(c)^0} \quad \dfrac{0 \quad \dfrac{b^{-1}}{(b)^0} \quad \dfrac{a^{-1}}{(a)^0}}{(a \bullet b)^0}}{((a \bullet b) \bullet c)^0}$$

$$a \bullet (b \bullet c) \to (a \bullet b) \bullet c$$

$$a \cdot b \cdot c \cdot 1 \cdot 1 \cdot 0 \cdot c^{-1} \cdot 0 \cdot b^{-1} \cdot a^{-1} \neq e$$

Failure of associativity: modal L

$$\dfrac{\dfrac{\dfrac{a}{{}^1(a)} \quad \dfrac{\dfrac{\dfrac{b}{{}^1(b)} \quad \dfrac{c}{{}^1(c)}}{(b \bullet c)^1} \quad 1}{{}^1(\Diamond(b \bullet c))}}{(a \bullet \Diamond(b \bullet c))^1} \quad 1}{{}^1(\Diamond(a \bullet \Diamond(b \bullet c)))} \qquad \dfrac{0 \quad \dfrac{\dfrac{c^{-1}}{(c)^0} \quad \dfrac{0 \quad \dfrac{\dfrac{b^{-1}}{(b)^0} \quad \dfrac{a^{-1}}{(a)^0}}{{}^0(a \bullet b)}}{(\Diamond(a \bullet b))^0}}{{}^0(\Diamond(a \bullet b) \bullet c)}}{(\Diamond(\Diamond(a \bullet b) \bullet c))^0}$$

$$\Diamond(a \bullet \Diamond(b \bullet c)) \to \Diamond(\Diamond(a \bullet b) \bullet c)$$

$$a \cdot b \cdot c \cdot 1 \cdot 1 \cdot 0 \cdot c^{-1} \cdot 0 \cdot b^{-1} \cdot a^{-1} \neq e$$

Failure of associativity: NL, Geach

$$\dfrac{\dfrac{a}{{}^1(a)} \quad 0 \quad \dfrac{b^{-1}}{(b)^0}}{{}^1(a/b)} \qquad \dfrac{\dfrac{\dfrac{b}{{}^1(b)} \quad 0 \quad \dfrac{c^{-1}}{(c)^0}}{{}^1(b/c)} \quad 1 \quad \dfrac{\dfrac{c}{{}^1(c)} \quad 1 \quad \dfrac{a^{-1}}{(a)^0}}{(a/c)^0}}{((a/c)/(b/c))^0}$$

$$a/b \to (a/c)/(b/c)$$

$$a \cdot 0 \cdot b^{-1} \cdot b \cdot 0 \cdot c^{-1} \cdot 1 \cdot c \cdot 1 \cdot a^{-1} \neq e$$

Failure of associativity (Geach): modal L

$$\dfrac{\dfrac{\dfrac{a}{{}^1(a)} \quad 0}{(\Box^{\downarrow}a)^1} \quad \dfrac{b^{-1}}{(b)^0}}{{}^1(\Box^{\downarrow}a/b)} \qquad \dfrac{\dfrac{\dfrac{\dfrac{b}{{}^1(b)} \quad 0}{(\Box^{\downarrow}b)^1} \quad \dfrac{c^{-1}}{(c)^0}}{{}^1(\Box^{\downarrow}b/c)} \quad \dfrac{1 \quad \dfrac{\dfrac{c}{{}^1(c)} \quad \dfrac{1 \quad \dfrac{a^{-1}}{(a)^0}}{{}^0(\Box^{\downarrow}a)}}{(\Box^{\downarrow}a/c)^0}}{{}^0(\Box^{\downarrow}(\Box^{\downarrow}a/c))}}{(\Box^{\downarrow}(\Box^{\downarrow}a/c)/(\Box^{\downarrow}b/c))^0}$$

$$\Box^{\downarrow}a/b \to \Box^{\downarrow}(\Box^{\downarrow}a/c)/(\Box^{\downarrow}b/c)$$

$$a \cdot 0 \cdot b^{-1} \cdot b \cdot 0 \cdot c^{-1} \cdot 1 \cdot c \cdot 1 \cdot a^{-1} \neq e$$

References

[1] Abrusci, M., C. Casadio and M. Moortgat (eds) (1994) *Linear Logic and Lambek Calculus.* Proceedings 1st Rome Workshop, June 1993. OTS/DYANA, Utrecht, Amsterdam.

[2] Benthem, J. van (1983) 'The semantics of variety in categorial grammar'. Report 83–29, Math Dept. Simon Fraser University, Burnaby, Canada.

[3] Benthem, J. van (1991) *Language in Action. Categories, Lambdas, and Dynamic Logic.* Studies in Logic, North-Holland, Amsterdam.

[4] Benthem, J. van (1984) 'Correspondence theory'. In Gabbay and Guenthner (eds) *Handbook of Philosophical Logic. Vol II.* Dordrecht, pp 167–247.

[5] Belnap, N.D. (1982) 'Display Logic'. *Journal of Philosophical Logic*, **11**, 375–417.

[6] Blyth, T.S. and M.F. Janowitz (1972) *Residuation Theory.* New York.

[7] Buszkowski, W. (1988) 'Generative power of categorial grammars'. In Oehrle, Bach and Wheeler (eds) *Categorial Grammars and Natural Language Structures.* Dordrecht.

[8] Došen, K. (1992) 'A brief survey of frames for the Lambek calculus'. Zeitschr. f. math. Logik und Grundlagen d. Mathematik **38**, 179–187.

[9] Došen, K. (1988,1989) 'Sequent systems and groupoid models'. *Studia Logica* **47**, 353–385, **48**, 41–65.

[10] Dunn, J.M. (1991) 'Gaggle theory: an abstraction of Galois connections and residuation, with applications to negation, implication, and various logical operators'. In Van Eijck (ed.) *Logics in AI.* JELIA Proceedings. Springer, Berlin.

[11] Dunn, M. (1993) 'Partial Gaggles Applied to Logics With Restricted Structural Rules'. In Došen and Schröder-Heister (eds) *Substructural Logics.* Oxford.

[12] Fuchs, L. (1963) *Partially-ordered Algebraic Systems.* New York.

[13] Gabbay, D. (1991) *Labelled Deductive Systems.* Draft. Oxford University Press (to appear).

[14] Gabbay, D. (1993) 'A general theory of structured consequence relations'. In Došen and Schröder-Heister (eds) *Substructural Logics.* Oxford.

[15] Hendriks, H. (1993) *Studied Flexibility.* PhD Dissertation, ILLC, Amsterdam.

[16] Hepple, M. (1990) *The Grammar and Processing of Order and Dependency: A Categorial Approach.* Ph.D. Dissertation, Edinburgh.

[17] Hepple, M. (1994) 'Labelled deduction and discontinuous constituency'. In [1].

[18] Janssen, Th. (1991) 'On properties of the Zielonka-Lambek calculus'. In Dekker and Stokhof (eds) *Proceedings 8th Amsterdam Colloquium.* ILLC, Amsterdam, pp 303–308.

[19] Kandulski, W. (1988) 'The non-associative Lambek calculus'. In Buszkowski, Marciszewski and van Benthem (eds) *Categorial Grammar.* Amsterdam.

[20] Kurtonina, N. (1994) 'On some modal extensions of the Lambek Calculus'. In [1].

[21] Lambek, J. (1958) 'The Mathematics of Sentence Structure', *American Mathematical Monthly* **65**, 154–170.

[22] Lambek, J. (1988) 'Categorial and categorical grammar'. In Oehrle, Bach and Wheeler (eds) *Categorial Grammars and Natural Language Structures*. Dordrecht.

[23] Moortgat, M. and G. Morrill (1991) 'Heads and phrases. Type calculus for dependency and constituent structure'. Ms OTS Utrecht.

[24] Moortgat, M. and R. Oehrle (1993) *Logical parameters and linguistic variation. Lecture notes on categorial grammar*. 5th European Summer School in Logic, Language and Information. Lisbon.

[25] Moortgat, M. and R.T. Oehrle (1994) 'Adjacency, dependency and order'. Proceedings 9th Amsterdam Colloquium, pp 447–466.

[26] Morrill, G. (1990) 'Intensionality and boundedness'. *L& P* **13**, pp 699–726.

[27] Morrill, G. (1992) 'Categorial formalisation of relativisation: pied-piping, islands and extraction sites'. Report LSI–92–23–R, Universitat Politècnica de Catalunya.

[28] Morrill, G. (1994) 'Structural facilitation and structural inhibition'. In [1].

[29] Pentus, M. (1992) 'Equivalent types in Lambek Calculus and Linear Logic'. MIAN Prepublication LCS–92–02. To appear in JoLLI.

[30] Roorda, D. (1991) *Resource logics. Proof-theoretical investigations*. Ph.D. Dissertation, ILLC, Amsterdam.

[31] Routley, R. and R.K. Meyer (1972–73) 'The semantics of entailment I'. In H.Leblanc (ed) *Truth, Syntax, Modality*, North-Holland, Amsterdam, pp 199–243.

[32] Steedman, M. (1993) 'Categorial grammar. Tutorial overview'. *Lingua* **90**, 221-258.

[33] Venema, Y. (1993) 'Meeting strength in substructural logics'. UU Logic Preprint. To appear in Studia Logica.

[34] Versmissen, K. (1993) 'Categorial grammar, modalities and algebraic semantics'. Proceedings EACL93, pp 377–383.

[35] Wallen, L.A. (1990) *Automated Deduction in Nonclassical Logics*. MIT Press, Cambridge, London.