

Proof Nets For Display Logic

Richard Moot

Workshop in Honor of Matteo Capelletti's defense

7 July 2007

Introduction

Beyond the non-associative Lambek calculus

- In the last decade, many new logical connectives have been added to the (non-associative) Lambek calculus **NL**.
- They include: unary modalities \diamond and \square , Galois $^{\perp}$, $^{\cdot\perp}$ and dual Galois \perp_{\cdot} , \perp_{\cdot} connections and a dual residuated family of connectives \multimap , \wp , \leftarrow .
- Proof theory has not kept up with these new developments.
- We would like to have a non-redundant proof system similar to the proof nets for linear logic.

Introduction

Proof Nets for Display Logic

- By reflecting on the nature of the objects in our universe of links — our basic building blocks for constructing proof nets — we can incorporate all of the proposed extensions in one proof net calculus.
- We will look at this calculus as well as basic soundness and completeness results.
- I will finish by analysing some properties of the Lambek-Grishin calculus **LG** and sketch some new applications.

Links

The Anatomy of a Link

Definition

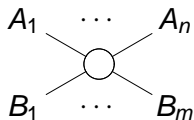
A *link* is a tuple $\langle \tau, P, Q, m \rangle$ where

- τ , the type of the link, is either \otimes or \wp ,
- P is a list of premisses A_1, \dots, A_n ,
- Q is a list of conclusions B_1, \dots, B_m ,
- m , the main formula of the link, is either ϵ or a member of $P \cup Q$.

If $m = \epsilon$ then we will call the link *neutral*, if it is a member of P we will call the link a *left link* and if it is a member of Q we will call it a *right link*.

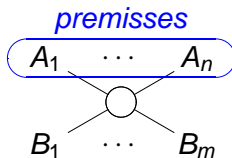
Links

The Anatomy of a Link



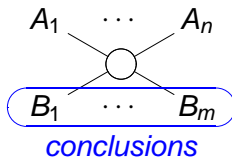
Links

The Anatomy of a Link



Links

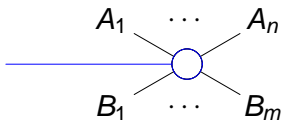
The Anatomy of a Link



Links

The Anatomy of a Link

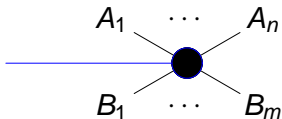
Link type: tensor



Links

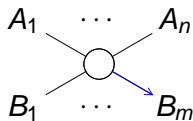
The Anatomy of a Link

Link type: par



Links

The Anatomy of a Link



Main formula

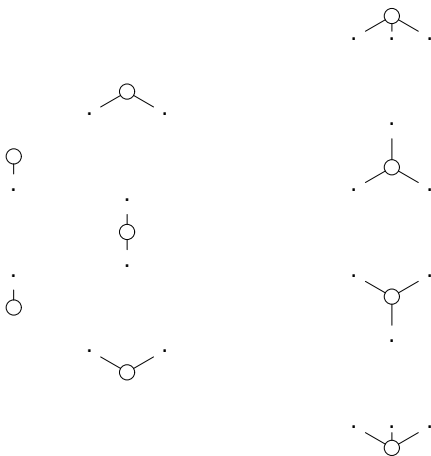
Links

Tensor Links: Different Shapes

Nullary

Unary

Binary



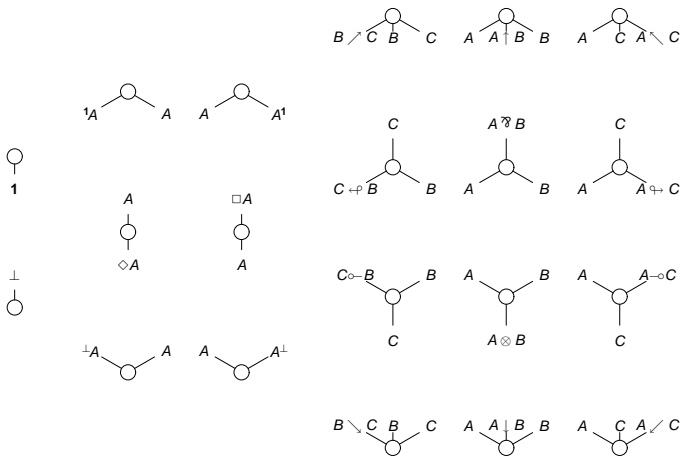
Links

Tensor Links

Nullary

Unary

Binary

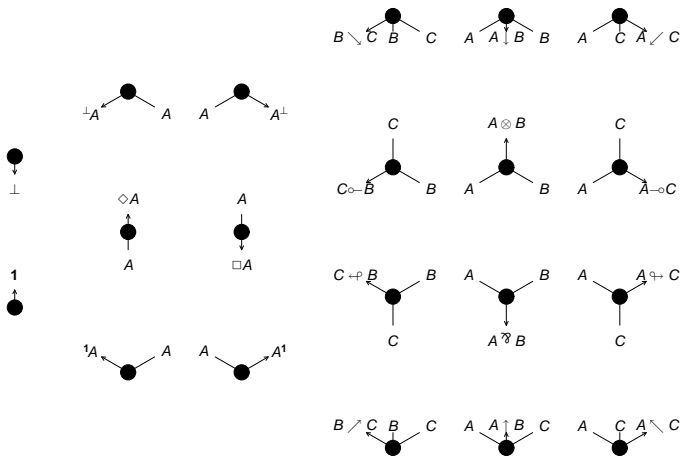


Links
Par Links

Nullary

Unary

Binary



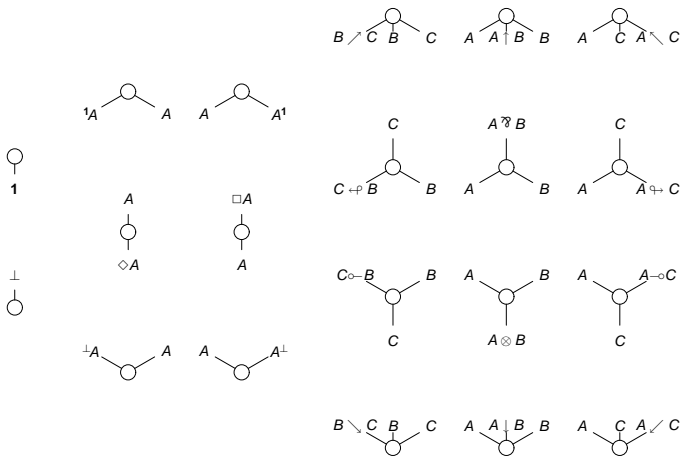
Links

Some interesting special cases

Nullary

Unary

Binary

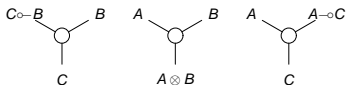


Links

Some interesting special cases

Unary

Binary

NL

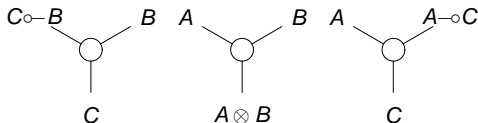
Links

Some interesting special cases

Unary

Binary

NL

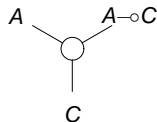
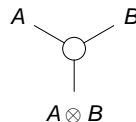
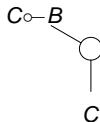
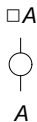
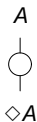


Links

Some interesting special cases

Unary

Binary

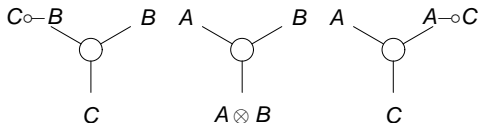
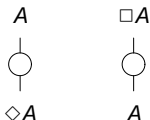
 $NL \diamond$ 

Links

Some interesting special cases

Unary

Binary

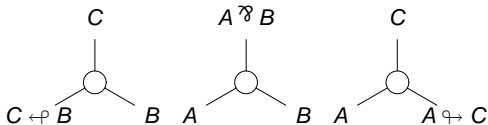
NL \diamond 1

Links

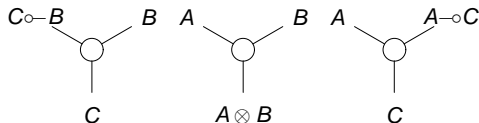
Some interesting special cases

Unary

Binary



GL



Proof Structures

Definition

Definition

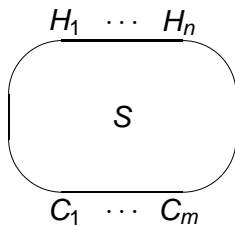
A *proof structure* $\langle S, \mathcal{L} \rangle$ is a finite set of formulas S together with a set of links \mathcal{L} as shown in [tensor table](#) and [par table](#) such that.

- every formula of S is at most once the premiss of a link.
- every formula of S is at most once the conclusion of a link.

Formulas which are not the conclusion of any link are the *hypotheses* H of the proof structures, whereas the formulas which are not the premiss of any link are the *conclusions* C .

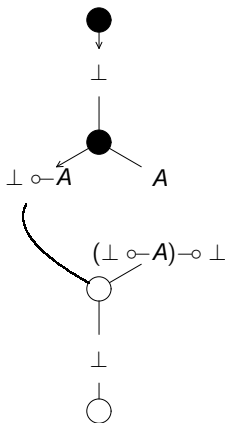
Proof Structures

General Form



Proof Structures

Example



Abstract Proof Structures

Definition

Definition

An *abstract proof structure* is a tuple $\langle V, \mathcal{L}, p, q \rangle$ such that.

V is a finite set of vertices,

\mathcal{L} is a set of links such that

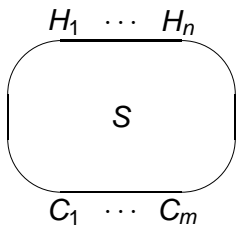
- every vertex of V is at most once the premiss of a link,
- every vertex of V is at most once the conclusion of a link

p is a labelling function assigning a formula to the hypotheses of the abstract proof structure, that is, to those formulas which are not the conclusion of any link,

q is a labelling function assigning a formula to the conclusions of the abstract proof structure, that is, to those formulas which are not the premiss of any link.

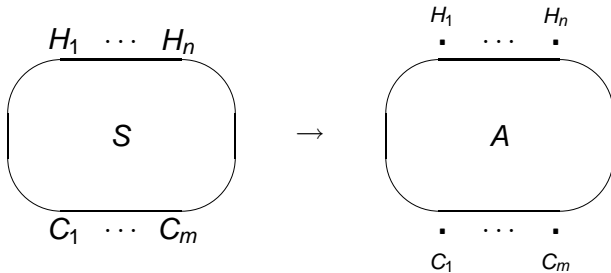
Abstract Proof Structures

General Form



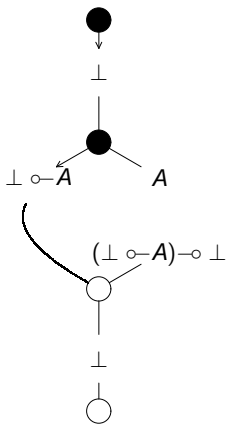
Abstract Proof Structures

General Form



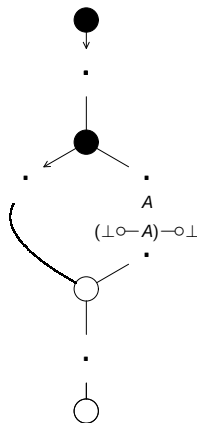
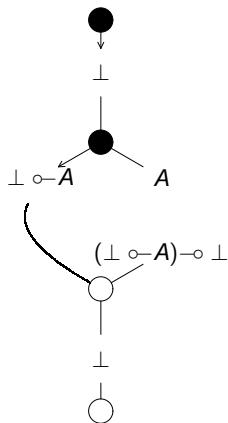
Abstract Proof Structures

Example



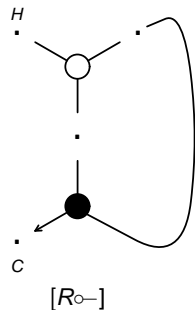
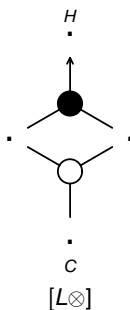
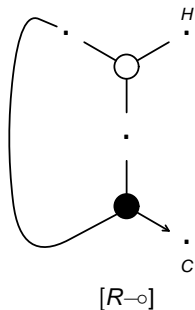
Abstract Proof Structures

Example



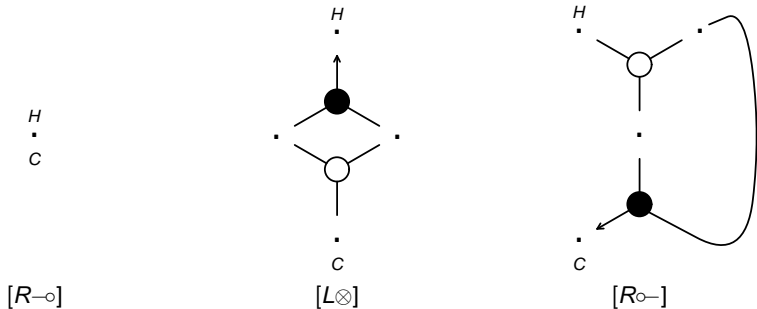
Contractions

Binary Residuated



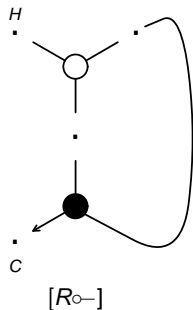
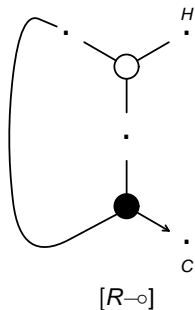
Contractions

Binary Residuated



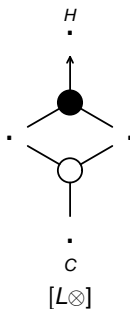
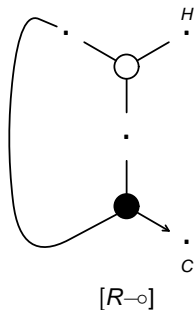
Contractions

Binary Residuated



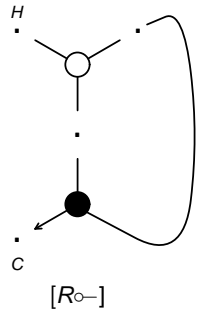
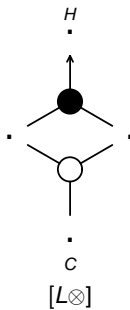
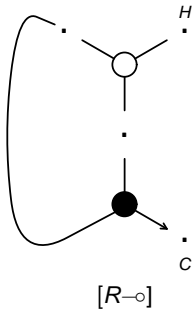
Contractions

Binary Residuated



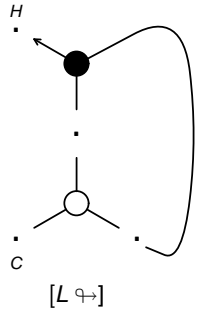
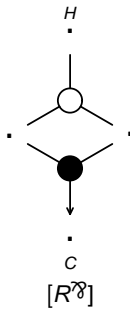
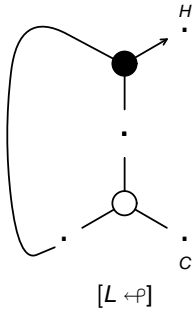
Contractions

Binary Residuated



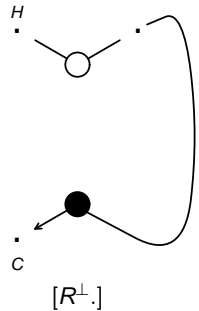
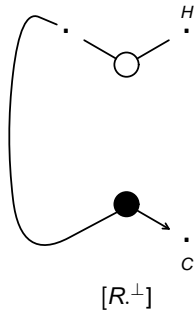
Contractions

Binary Dual Residuated



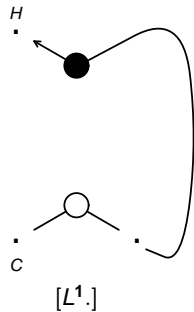
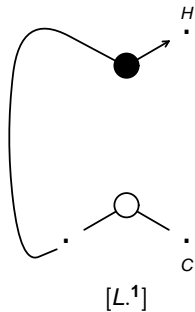
Contractions

Unary 1



Contractions

Unary 2

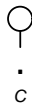


Contractions

Nullary



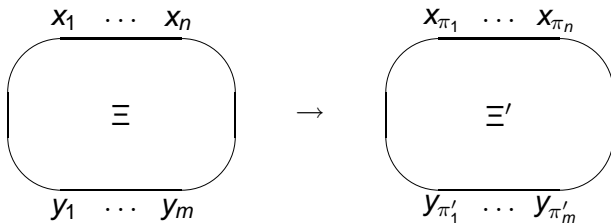
$[R \perp]$



$[L1]$

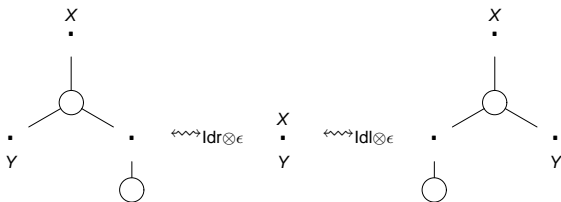
Structural Rules

Schematic Form



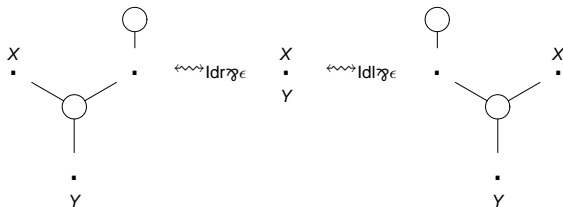
Structural Rules

Examples: Identity



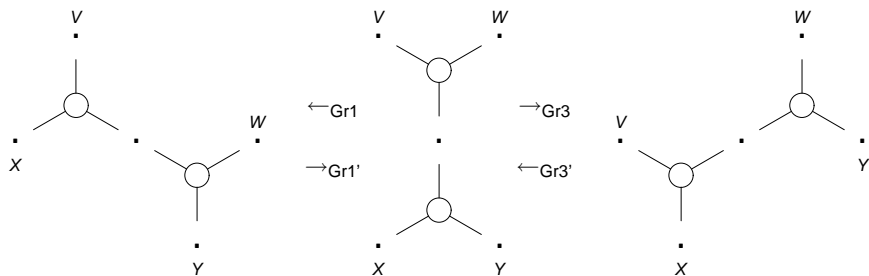
Structural Rules

Examples: Identity



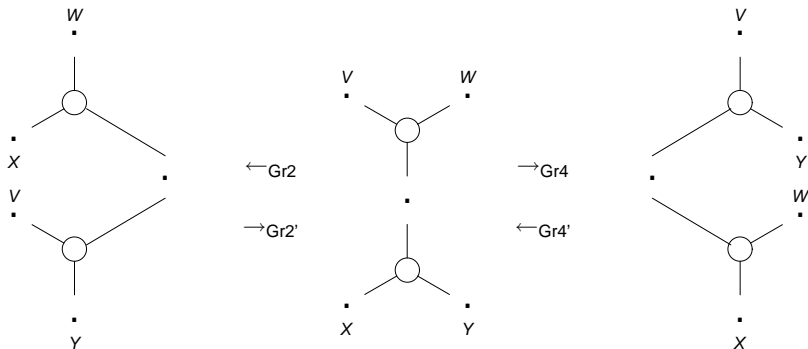
Structural Rules

Examples: Grishin



Structural Rules

Examples: Grishin (contd.)



Proof Nets

Definition

Definition

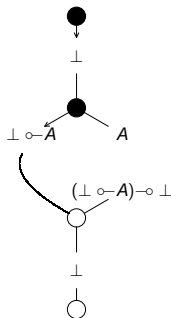
A *tensor tree* is an acyclic connected abstract proof structure containing only tensor links.

Definition

A proof structure is a *proof net* iff it abstract proof structure converts to a tensor tree using the structural conversions and the contractions.

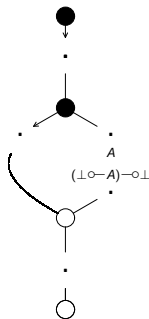
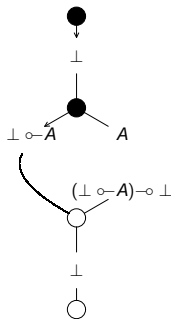
Proof Nets

Example: Proof Structure and Abstract Proof Structure



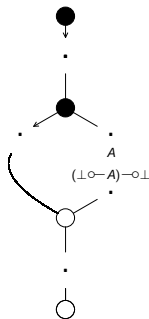
Proof Nets

Example: Proof Structure and Abstract Proof Structure



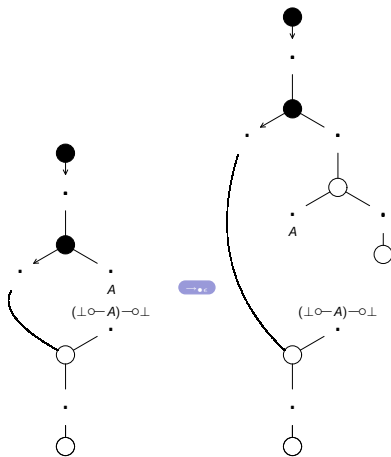
Proof Nets

Example (contd.)



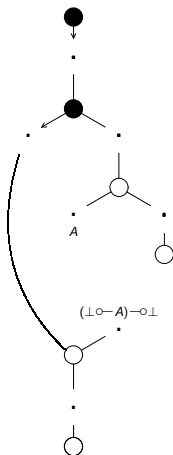
Proof Nets

Example (contd.)



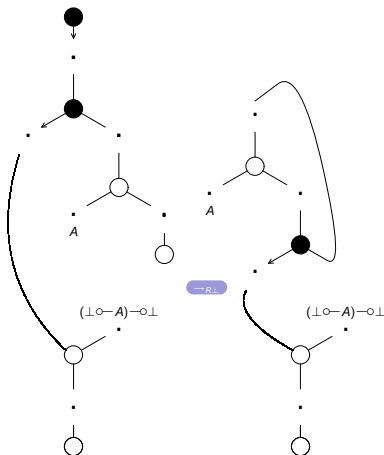
Proof Nets

Example (contd.)



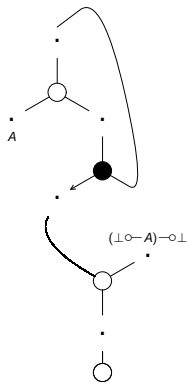
Proof Nets

Example (contd.)



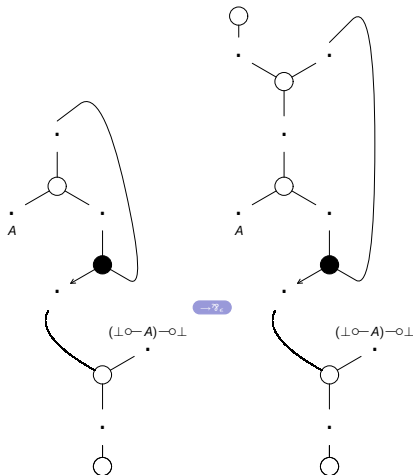
Proof Nets

Example (contd.)



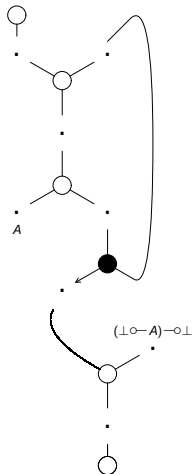
Proof Nets

Example (contd.)



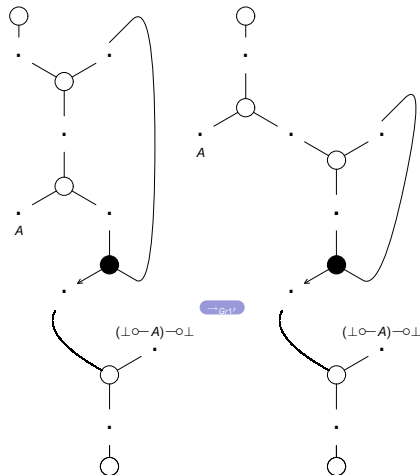
Proof Nets

Example (contd.)



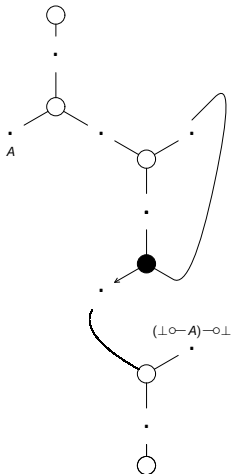
Proof Nets

Example (contd.)



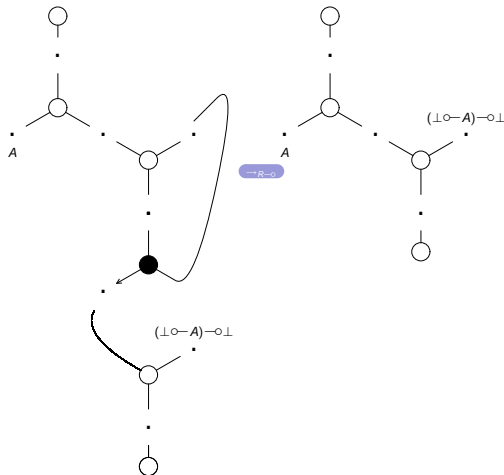
Proof Nets

Example (contd.)



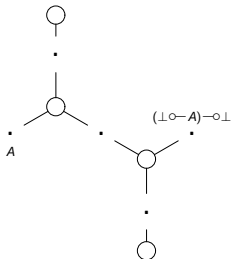
Proof Nets

Example (contd.)



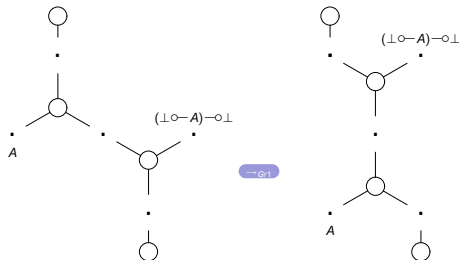
Proof Nets

Example (contd.)



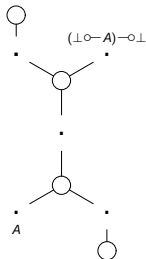
Proof Nets

Example (contd.)



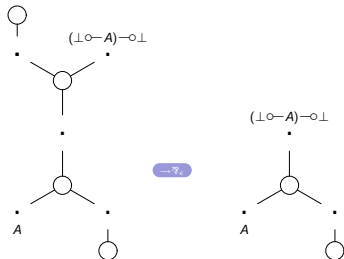
Proof Nets

Example (contd.)



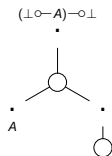
Proof Nets

Example (contd.)



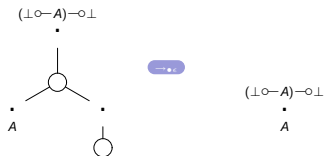
Proof Nets

Example (contd.)



Proof Nets

Example (contd.)



Soundness and Completeness

Theorem

Theorem

A proof structure S is correct (ie. corresponds to a sequent proof of $\Gamma \vdash \Delta$) if and only if its abstract proof structure A converts to a tensor tree of $\Gamma \vdash \Delta$ (ie. S is a proof net).

Soundness and Completeness

Soundness: Axioms

We proceed by induction on the depth of the sequent calculus proof. In case $d = 1$ we have [one of the axioms](#). We verify all of them are proof nets.

$$A \vdash A$$

Soundness and Completeness

Soundness: Axioms

We proceed by induction on the depth of the sequent calculus proof. In case $d = 1$ we have [one of the axioms](#). We verify all of them are proof nets.

$$A \vdash A$$

$$A \quad \rightarrow \quad \begin{array}{c} A \\ \cdot \\ A \end{array}$$

Soundness and Completeness

Soundness: Axioms

We proceed by induction on the depth of the sequent calculus proof. In case $d = 1$ we have [one of the axioms](#). We verify all of them are proof nets.

$$\epsilon \vdash \mathbf{1}$$

Soundness and Completeness

Soundness: Axioms

We proceed by induction on the depth of the sequent calculus proof. In case $d = 1$ we have **one of the axioms**. We verify all of them are proof nets.

$$\epsilon \vdash \mathbf{1}$$



Soundness and Completeness

Soundness: Axioms

We proceed by induction on the depth of the sequent calculus proof. In case $d = 1$ we have [one of the axioms](#). We verify all of them are proof nets.

$$\perp \vdash \epsilon$$

Soundness and Completeness

Soundness: Axioms

We proceed by induction on the depth of the sequent calculus proof. In case $d = 1$ we have [one of the axioms](#). We verify all of them are proof nets.

$$\perp \vdash \epsilon$$



Soundness and Completeness

Soundness: Inductive Case

Depth $d > 1$. We look at the last rule.

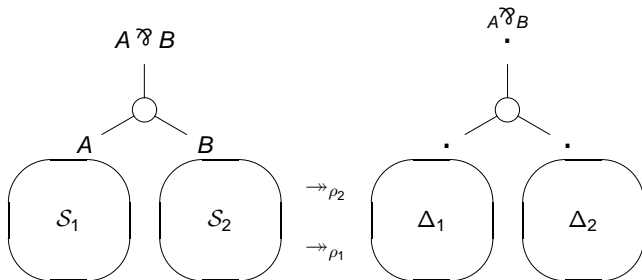
$$\frac{\begin{array}{c} \vdots \pi_1 \\ A \vdash \Delta_1 \end{array} \quad \begin{array}{c} \vdots \pi_2 \\ B \vdash \Delta_2 \end{array}}{A \wp B \vdash \Delta_1 \circ \Delta_2} [L\wp]$$

Soundness and Completeness

Soundness: Inductive Case

Depth $d > 1$. We look at the last rule.

$$\frac{\begin{array}{c} \vdots \pi_1 \\ A \vdash \Delta_1 \end{array} \quad \begin{array}{c} \vdots \pi_2 \\ B \vdash \Delta_2 \end{array}}{A \wp B \vdash \Delta_1 \circ \Delta_2} [L\wp]$$



Soundness and Completeness

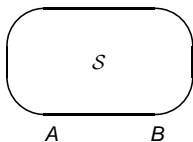
Soundness: Inductive Case

$$\frac{\Gamma \vdash A \circ B}{\Gamma \vdash A \wp B} [R\wp]$$

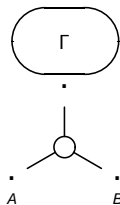
Soundness and Completeness

Soundness: Inductive Case

$$\frac{\Gamma \vdash A \circ B}{\Gamma \vdash A \wp B} [R\wp]$$



$\rightarrow_{\rho 1}$

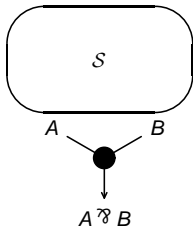


Soundness and Completeness

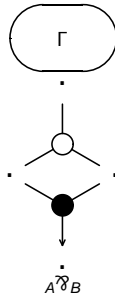
Soundness: Inductive Case

$$\frac{\Gamma \vdash A \circ B}{\Gamma \vdash A \wp B} [R\wp]$$

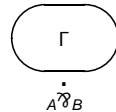
$\vdots \pi_1$



$\rightarrow_{\rho 1}$



$\rightarrow_{R\wp}$



Soundness and Completeness

Completeness: Empty conversion sequence

We proceed by induction on the length l of the conversion sequence. In case $l = 0$ then \mathcal{S} corresponds to a tensor tree and we proceed by induction on the number of tensor links t .

$$t = 0$$

Soundness and Completeness

Completeness: Empty conversion sequence

We proceed by induction on the length l of the conversion sequence. In case $l = 0$ then \mathcal{S} corresponds to a tensor tree and we proceed by induction on the number of tensor links t .

$$t = 0$$

$$A \rightarrow \begin{array}{c} A \\ \cdot \\ A \end{array}$$

Soundness and Completeness

Completeness: Empty conversion sequence

$$t > 0$$

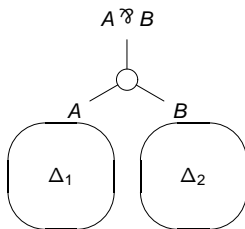
We find the principal leaf.

Soundness and Completeness

Completeness: Empty conversion sequence

$$t > 0$$

We find the principal leaf.

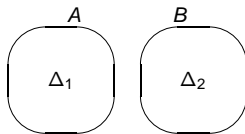


Soundness and Completeness

Completeness: Empty conversion sequence

$$t > 0$$

We find the principal leaf.



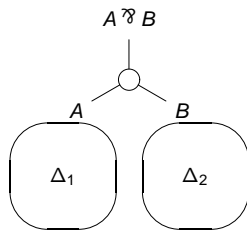
Soundness and Completeness

Completeness: Empty conversion sequence

$$t > 0$$

We find the principal leaf.

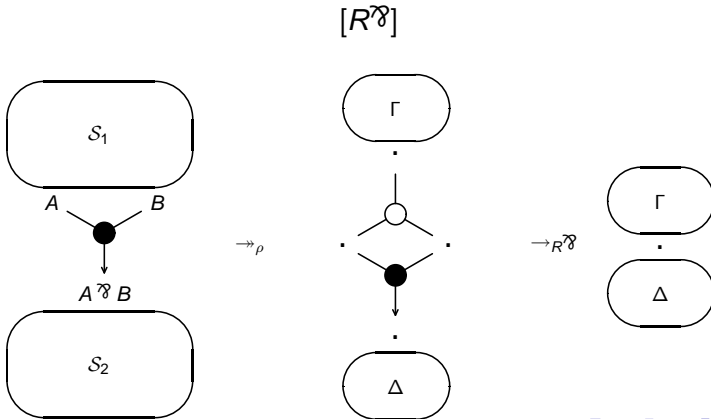
$$\frac{A \vdash \Delta_1 \quad B \vdash \Delta_2}{A \wp B \vdash \Delta_1 \circ \Delta_2} [L\wp]$$



Soundness and Completeness

Completeness: Inductive Case

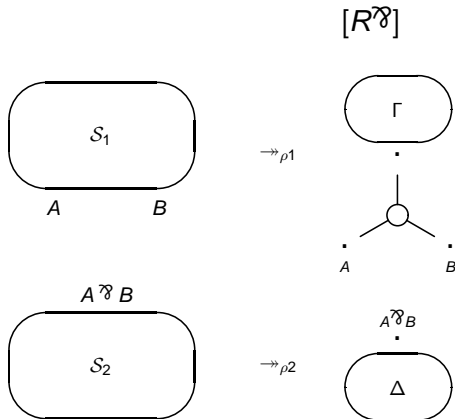
We look at the last conversion.



Soundness and Completeness

Completeness: Inductive Case

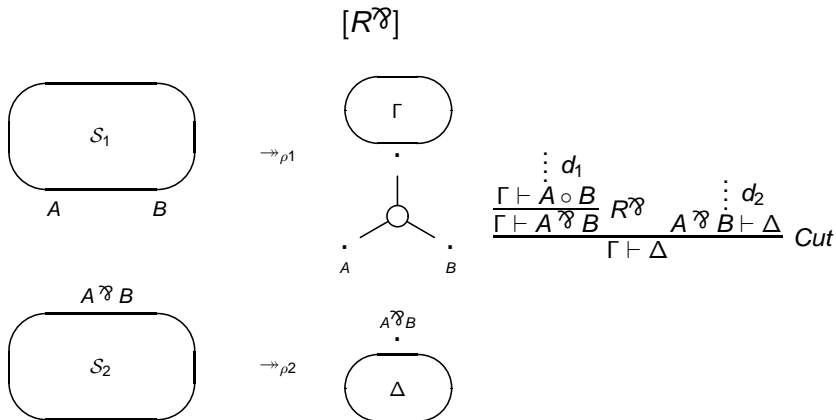
We look at the last conversion.



Soundness and Completeness

Completeness: Inductive Case

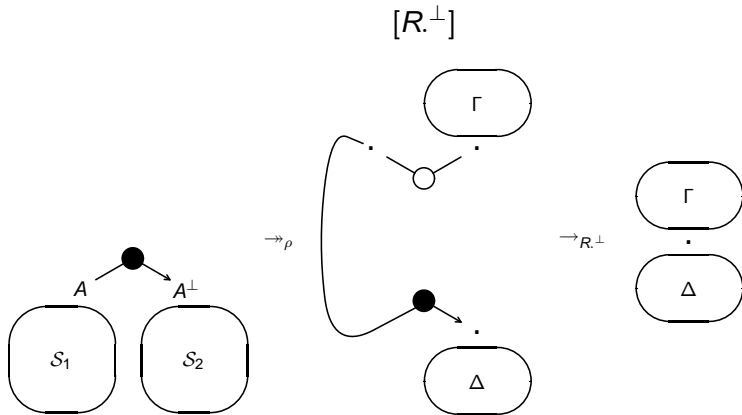
We look at the last conversion.



Soundness and Completeness

Completeness: Inductive Case

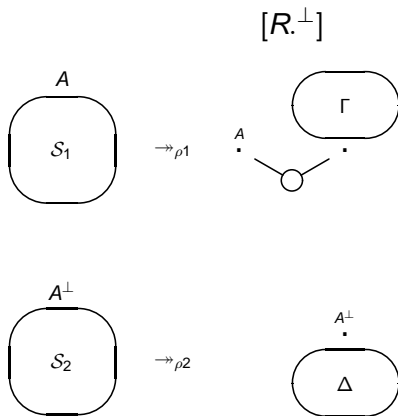
We look at the last conversion.



Soundness and Completeness

Completeness: Inductive Case

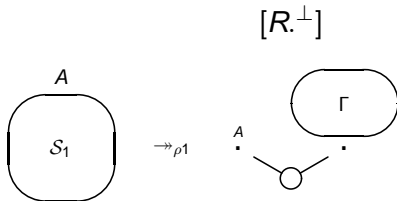
We look at the last conversion.



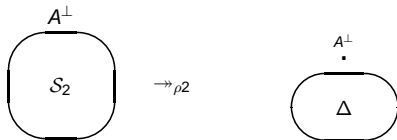
Soundness and Completeness

Completeness: Inductive Case

We look at the last conversion.



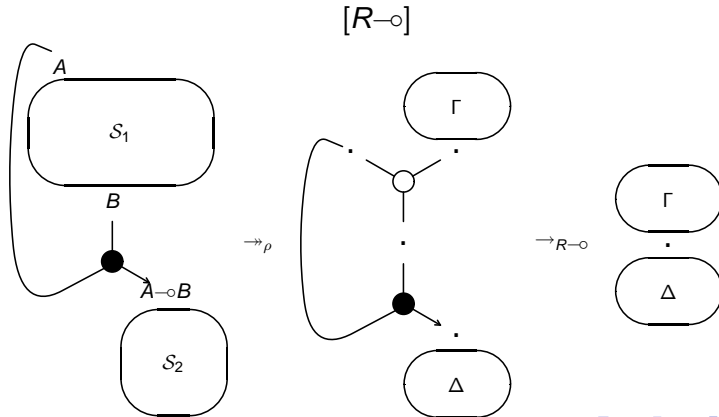
$$\frac{\frac{\Gamma \vdash [A]}{\Gamma \vdash A^\perp} R.\perp \quad \frac{\vdots d_1 \quad \vdots d_2}{A^\perp \vdash \Delta} Cut}{\Gamma \vdash \Delta} Cut$$



Soundness and Completeness

Completeness: Inductive Case

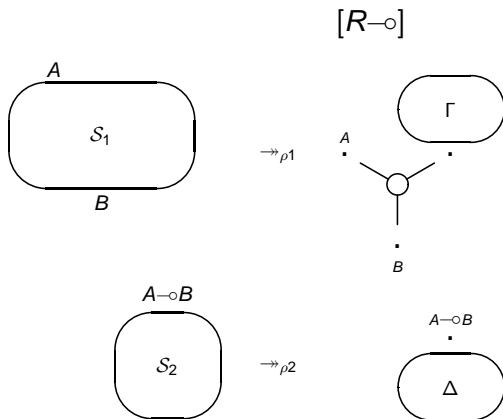
We look at the last conversion.



Soundness and Completeness

Completeness: Inductive Case

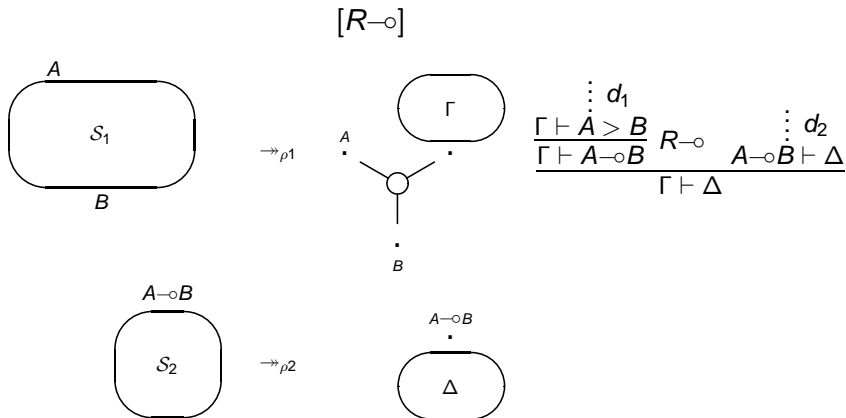
We look at the last conversion.



Soundness and Completeness

Completeness: Inductive Case

We look at the last conversion.

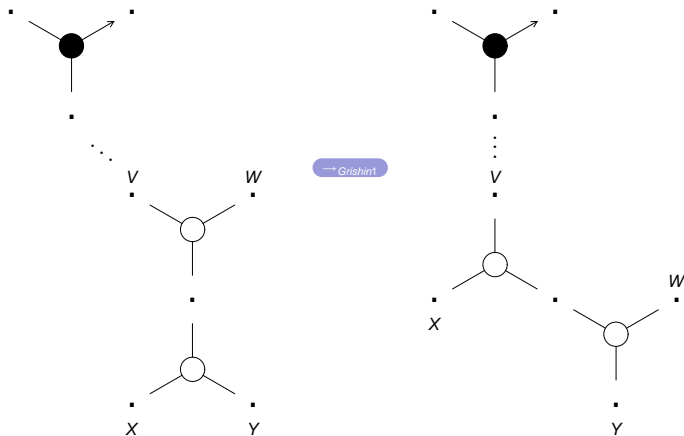


Applications

- Now that we have the basic results in place, we can reflect on some of the possible applications. Here we will focus on the Lambek-Grishin calculus **LG**, which has the **links**, **contractions** and **structural rules** we've seen before.
- Moortgat gives type assignments for *quantifier scope* as well as for *wh extraction*.
- We will give two new applications by giving proof nets which can embed *tree adjoining grammars* and handle *scrambling*.
- Be first we will look at the interaction between the contractions and the structural rules.

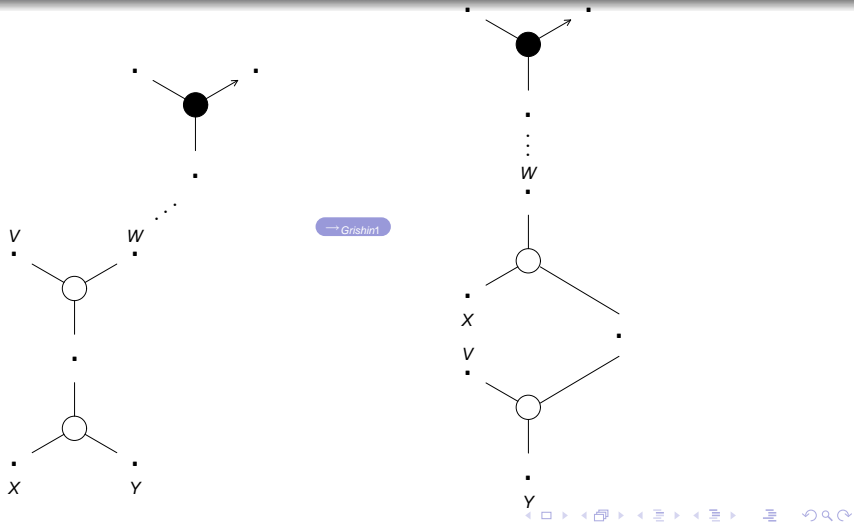
Grishin Contractions

Left: Grishin 1



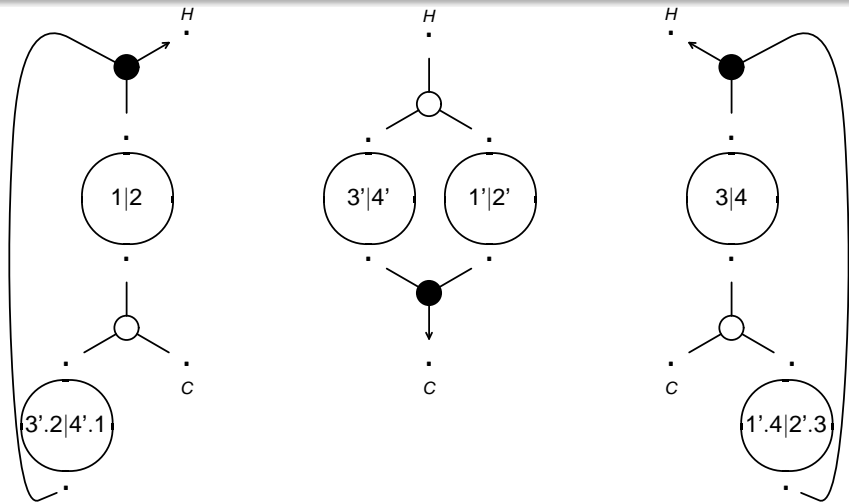
Grishin Contractions

Right: Grishin 2



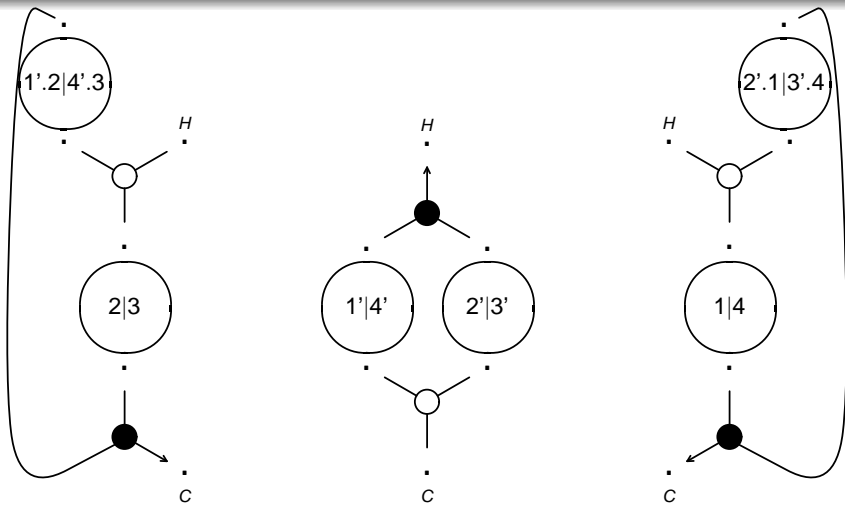
Contractions for the Lambek-Grishin Calculus

Dual Residuated Contractions With Grishin Rules



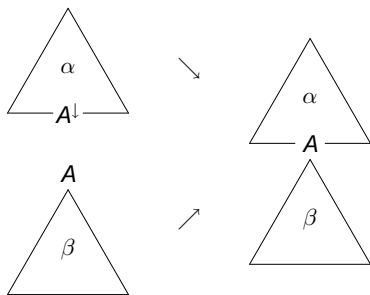
Contractions for the Lambek-Grishin Calculus

Residuated Contractions With Grishin Rules



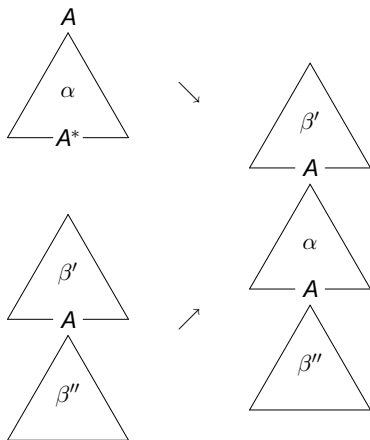
Tree Adjoining Grammars

Definition: Substitution



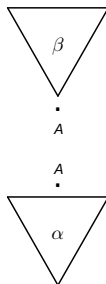
Tree Adjoining Grammars

Definition: Adjunction



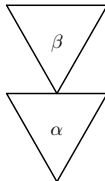
Tree Adjoining Grammars

Substitution



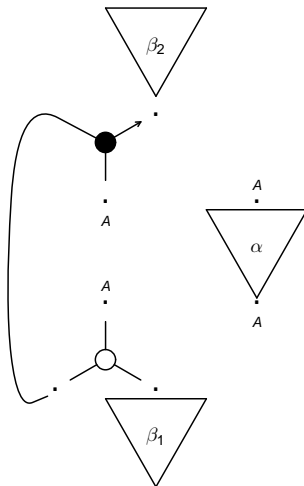
Tree Adjoining Grammars

Substitution: Axiom



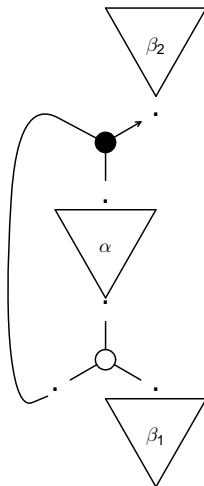
Tree Adjoining Grammars

Adjunction



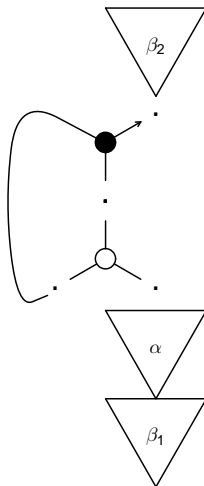
Tree Adjoining Grammars

Adjunction: Axiom



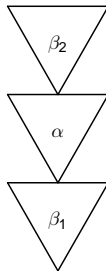
Tree Adjoining Grammars

Adjunction: Grishin 1,2



Tree Adjoining Grammars

Adjunction: Contraction



Scrambling

Non-determinism in the Contractions

- Looking back at the **LG contractions** we see that there are always *two* possibilities for interaction with the Grishin rules.
- This introduces non-determinism in the computation, allowing us to ‘move’ the sister of the top node of either of the two subnets in the Grishin contractions.
- This means that if we have two completely left-branching trees we can interleave all leaves of the two trees.

Scrambling

Draw Example on the Board

...

Conclusions and Open Questions

Conclusions

- A simple change of perspective on the type of links which inhabit our universe opens the door to many new logical connectives.
- Soundness and completeness results incorporate these changes without problems.
- Some new applications have been suggested.

Conclusions and Open Questions

Open Questions

- What is the computational complexity of **LG**?
- What class of languages does **LG** generate?
- What is the influence of the 8 different Grishin structural rules on the complexity/class of languages?

Appendix

Sequent Rules

Sequent Rules

Sequent Rules

Display Rules

$$\frac{[\Gamma] \vdash \Delta}{[\Delta] \vdash \Gamma} [dgc]$$

$$\frac{\Gamma \vdash [\Delta]}{\Delta \vdash [\Gamma]} [gc]$$

$$\frac{\langle \Gamma \rangle \vdash \Delta}{\Gamma \vdash \langle \Delta \rangle} [rc]$$

$$\frac{\Gamma_1 \vdash \Delta < \Gamma_2}{\Gamma_1 \circ \Gamma_2 \vdash \Delta} [rc]$$

$$\frac{\Gamma_1 \circ \Gamma_2 \vdash \Delta}{\Gamma_2 \vdash \Gamma_1 > \Delta} [rc]$$

$$\frac{\Gamma < \Delta_2 \vdash \Delta_1}{\Gamma \vdash \Delta_1 \circ \Delta_2} [drc]$$

$$\frac{\Gamma \vdash \Delta_1 \circ \Delta_2}{\Delta_1 > \Gamma \vdash \Delta_2} [drc]$$

$$\frac{\Delta \triangleright \Gamma_2 \vdash \Gamma_1}{\Gamma_1 \diamond \Gamma_2 \vdash \Delta} [dgc]$$

$$\frac{\Gamma_1 \diamond \Gamma_2 \vdash \Delta}{\Gamma_1 \triangleright \Delta \vdash \Gamma_2} [dgc]$$

$$\frac{\Delta_1 \vdash \Gamma \triangleleft \Delta_2}{\Gamma \vdash \Delta_1 \diamond \Delta_2} [gc]$$

$$\frac{\Gamma \vdash \Delta_1 \diamond \Delta_2}{\Delta_2 \vdash \Delta_1 \triangleright \Gamma} [gc]$$

Sequent Rules

Identity and Nullary Connectives

$$\frac{}{A \vdash A} [Ax] \quad \frac{\Gamma \vdash A \quad A \vdash \Delta}{\Gamma \vdash \Delta} [Cut]$$

$$\frac{\epsilon \vdash \Delta}{\mathbf{1} \vdash \Delta} [L\mathbf{1}] \quad \frac{}{\epsilon \vdash \mathbf{1}} [R\mathbf{1}]$$

$$\frac{}{\perp \vdash \epsilon} [L\perp] \quad \frac{\Gamma \vdash \epsilon}{\Gamma \vdash \perp} [R\perp]$$

Sequent Rules

Unary Connectives

$$\frac{\Delta \vdash A}{A^\perp \vdash [\Delta]} [L.\perp] \qquad \frac{\Gamma \vdash [A]}{\Gamma \vdash A^\perp} [R.\perp]$$

$$\frac{\Delta \vdash A}{\perp A \vdash [\Delta]} [L^\perp.] \qquad \frac{\Gamma \vdash [A]}{\Gamma \vdash \perp A} [R^\perp.]$$

$$\frac{[A] \vdash \Delta}{A^1 \vdash \Delta} [L.^1] \qquad \frac{A \vdash \Gamma}{[\Gamma] \vdash A^1} [R.^1]$$

$$\frac{[A] \vdash \Delta}{^1 A \vdash \Delta} [L^1.] \qquad \frac{A \vdash \Gamma}{[\Gamma] \vdash ^1 A} [R^1.]$$

$$\frac{\langle A \rangle \vdash \Delta}{\diamond A \vdash \Delta} [L\diamond] \qquad \frac{\Gamma \vdash A}{\langle \Gamma \rangle \vdash \diamond A} [R\diamond]$$

$$\frac{A \vdash \Delta}{\square A \vdash \langle \Delta \rangle} [L\square] \qquad \frac{\Gamma \vdash \langle A \rangle}{\Gamma \vdash \square A} [R\square]$$

Sequent Rules

Binary Connectives

$$\frac{A \circ B \vdash \Delta}{A \otimes B \vdash \Delta} [L\otimes]$$

$$\frac{\Gamma_1 \vdash A \quad \Gamma_2 \vdash B}{\Gamma_1 \circ \Gamma_2 \vdash A \otimes B} [R\otimes]$$

$$\frac{\Gamma \vdash A \quad B \vdash \Delta}{A \multimap B \vdash \Gamma > \Delta} [L\multimap]$$

$$\frac{\Gamma \vdash A > B}{\Gamma \vdash A \multimap B} [R\multimap]$$

$$\frac{A \vdash \Delta \quad \Gamma \vdash B}{A \multimap B \vdash \Delta < \Gamma} [L\multimap]$$

$$\frac{\Gamma \vdash A < B}{\Gamma \vdash A \multimap B} [R\multimap]$$

$$\frac{A \vdash \Delta_1 \quad B \vdash \Delta_2}{A \wp B \vdash \Delta_1 \circ \Delta_2} [L\wp]$$

$$\frac{\Gamma \vdash A \circ B}{\Gamma \vdash A \wp B} [R\wp]$$

$$\frac{A > B \vdash \Delta}{A \wp B \vdash \Delta} [L\wp]$$

$$\frac{A \vdash \Delta \quad \Gamma \vdash B}{\Delta > \Gamma \vdash A \wp B} [R\wp]$$

$$\frac{A < B \vdash \Delta}{A \wp B \vdash \Delta} [L\wp]$$

$$\frac{\Gamma \vdash A \quad B \vdash \Delta}{\Gamma < \Delta \vdash A \wp B} [R\wp]$$

Sequent Rules

Binary Connectives (continued)

$$\frac{\Gamma_1 \vdash A \quad \Gamma_2 \vdash B}{A \downarrow B \vdash \Gamma_1 \diamond \Gamma_2} [L \downarrow]$$

$$\frac{\Gamma \vdash A \diamond B}{\Gamma \vdash A \downarrow B} [R \downarrow]$$

$$\frac{\Gamma_1 \vdash A \quad \Gamma_2 \vdash B}{A \swarrow B \vdash \Gamma_1 \triangleleft \Gamma_2} [L \swarrow]$$

$$\frac{\Gamma \vdash A \triangleleft B}{\Gamma \vdash A \swarrow B} [R \swarrow]$$

$$\frac{\Gamma_1 \vdash A \quad \Gamma_2 \vdash B}{A \searrow B \vdash \Gamma_1 \triangleright \Gamma_2} [L \searrow]$$

$$\frac{\Gamma \vdash A \triangleright B}{\Gamma \vdash A \searrow B} [R \searrow]$$

$$\frac{A \diamond B \vdash \Delta}{A \uparrow B \vdash \Delta} [L \uparrow]$$

$$\frac{A \vdash \Delta_1 \quad B \vdash \Delta_2}{\Delta_1 \diamond \Delta_2 \vdash A \uparrow B} [R \uparrow]$$

$$\frac{A \triangleleft B \vdash \Delta}{A \swarrow B \vdash \Delta} [L \swarrow]$$

$$\frac{A \vdash \Delta_1 \quad B \vdash \Delta_2}{\Delta_1 \triangleleft \Delta_2 \vdash A \swarrow B} [R \swarrow]$$

$$\frac{A \triangleright B \vdash \Delta}{A \searrow B \vdash \Delta} [L \searrow]$$

$$\frac{A \vdash \Delta_1 \quad B \vdash \Delta_2}{\Delta_1 \triangleright \Delta_2 \vdash A \searrow B} [R \searrow]$$