Identity of Proofs and Formulas using Proof-Nets in Multiplicative-Additive Linear Logic

Rémi Di Guardia

supervised by Olivier Laurent

23 September 2024

Cooking

Mousse Recipe

Ingredients: chocolate, eggs

1. Warm the chocolate
2. Beat the whites
3. Add the yolks in the chocolate
4. Add the whites in the mixture

Cooking

Mousse Recipe

Ingredients: chocolate, eggs

1. Get the eggs
2. Get the chocolate
3. Warm the chocolate
4. Separate the eggs
5. Beat the whites
6. Add the yolks in the chocolate
7. Add the whites in the mixture

Cooking

Mousse Recipe

Ingredients: chocolate, eggs

1. Get the eggs
2. Get the chocolate
3. Warm the chocolate
4. Separate the eggs
5. Beat the whites
6. Add the yolks in the chocolate
7. Add the whites in the mixture

Another? Mousse Recipe

Ingredients: chocolate, eggs

1. Get the eggs
4. Separate the eggs
5. Beat the whites
2. Get the chocolate
3. Warm the chocolate
6. Add the yolks in the chocolate
7. Add the whites in the mixture

Cooking

Mousse Recipe

Ingredients: chocolate, eggs

1. Get the eggs
2. Get the chocolate
3. Warm the chocolate
4. Separate the eggs
5. Beat the whites
6. Add the yolks in the chocolate
7. Add the whites in the mixture

Same Mousse Recipe

Ingredients: chocolate, eggs

1. Get the eggs
4. Separate the eggs
5. Beat the whites
2. Get the chocolate
3. Warm the chocolate
6. Add the yolks in the chocolate
7. Add the whites in the mixture

$\longrightarrow$ The order of *independent* steps is meaningless for the result!
Different writings for a unique recipe

Another representation of recipes

Mousse Recipe

Ingredients: chocolate, eggs

1. Get the eggs
2. Get the chocolate
3. Warm the chocolate
4. Separate the eggs
5. Beat the whites
6. Add the yolks in the chocolate
7. Add the whites in the mixture

Another representation of recipes

Mousse Recipe

Ingredients: chocolate, eggs

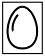
1. Get the eggs
2. Get the chocolate
3. Warm the chocolate
4. Separate the eggs
5. Beat the whites
6. Add the yolks in the chocolate
7. Add the whites in the mixture

Another representation of recipes

Mousse Recipe

Ingredients: chocolate, eggs

1. Get the eggs
2. Get the chocolate
3. Warm the chocolate
4. Separate the eggs
5. Beat the whites
6. Add the yolks in the chocolate
7. Add the whites in the mixture



Another representation of recipes

Mousse Recipe

Ingredients: chocolate, eggs

1. Get the eggs
2. Get the chocolate
3. Warm the chocolate
4. Separate the eggs
5. Beat the whites
6. Add the yolks in the chocolate
7. Add the whites in the mixture

Another representation of recipes

Mousse Recipe

Ingredients: chocolate, eggs

1. Get the eggs
2. Get the chocolate
3. Warm the chocolate
4. Separate the eggs
5. Beat the whites
6. Add the yolks in the chocolate
7. Add the whites in the mixture

Another representation of recipes

Mousse Recipe

Ingredients: chocolate, eggs

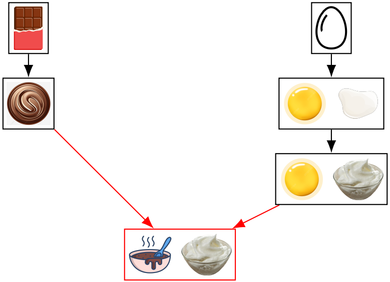
1. Get the eggs
2. Get the chocolate
3. Warm the chocolate
4. Separate the eggs
5. Beat the whites
6. Add the yolks in the chocolate
7. Add the whites in the mixture

Another representation of recipes

Mousse Recipe

Ingredients: chocolate, eggs

1. Get the eggs
2. Get the chocolate
3. Warm the chocolate
4. Separate the eggs
5. Beat the whites
6. Add the yolks in the chocolate
7. Add the whites in the mixture

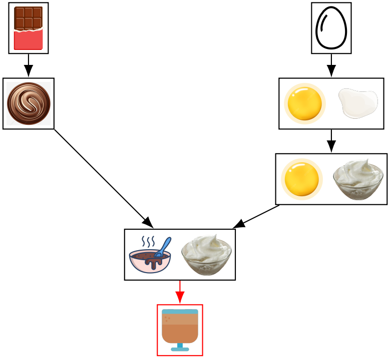


Another representation of recipes

Mousse Recipe

Ingredients: chocolate, eggs

1. Get the eggs
2. Get the chocolate
3. Warm the chocolate
4. Separate the eggs
5. Beat the whites
6. Add the yolks in the chocolate
7. Add the whites in the mixture

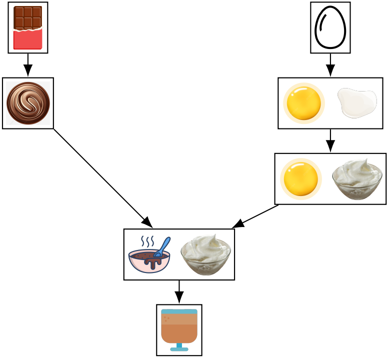


Another representation of recipes

Mousse Recipe

Ingredients: chocolate, eggs

1. Get the eggs
2. Get the chocolate
3. Warm the chocolate
4. Separate the eggs
5. Beat the whites
6. Add the yolks in the chocolate
7. Add the whites in the mixture



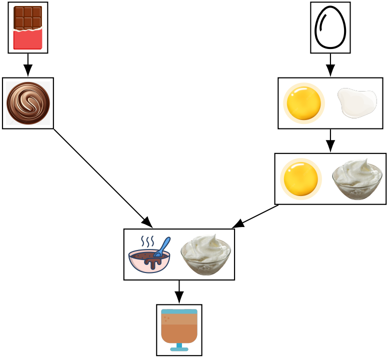
⟶ Better representation by following **causality**

Another representation of recipes

Same Mousse Recipe

Ingredients: chocolate, eggs

1. Get the eggs
4. Separate the eggs
5. Beat the whites
2. Get the chocolate
3. Warm the chocolate
6. Add the yolks in the chocolate
7. Add the whites in the mixture



⟶ Better representation by following **causality**

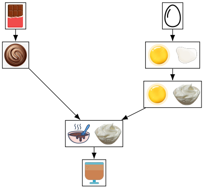
Diagrams solve the equality of recipes?

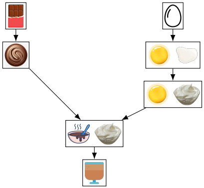
Recipe 1

1. Get eggs
2. Get chocolate
3. Warm chocolate
4. Separate eggs
5. Beat whites
6. Add yolks
7. Add whites

Recipe 2

1. Get eggs
4. Separate eggs
5. Beat whites
2. Get chocolate
3. Warm chocolate
6. Add yolks
7. Add whites





Diagrams solve the equality of recipes?

Recipe 1

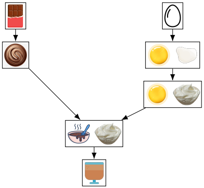
1 Get eggs
2 Get chocolate
3 Warm chocolate
4 Separate eggs
5 Beat whites
6 Add yolks
7 Add whites

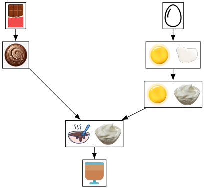
Recipe 2

1 Get eggs
4 Separate eggs
5 Beat whites
2 Get chocolate
3 Warm chocolate
6 Add yolks
7 Add whites

Recipe 3

1 Get eggs
2 Get chocolate
3 Warm chocolate
4 Separate eggs
5 Beat whites
7 Add whites
6 Add yolks





Diagrams solve the equality of recipes?

Recipe 1

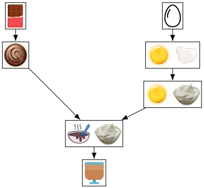
1 Get eggs
2 Get chocolate
3 Warm chocolate
4 Separate eggs
5 Beat whites
6 Add yolks
7 Add whites

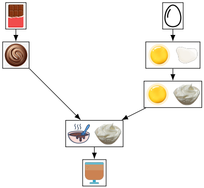
Recipe 2

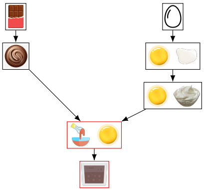
1 Get eggs
4 Separate eggs
5 Beat whites
2 Get chocolate
3 Warm chocolate
6 Add yolks
7 Add whites

Recipe 3

1 Get eggs
2 Get chocolate
3 Warm chocolate
4 Separate eggs
5 Beat whites
7 Add whites
6 Add yolks







Diagrams solve the equality of recipes?

Recipe 1

1 Get eggs
2 Get chocolate
3 Warm chocolate
4 Separate eggs
5 Beat whites
6 Add yolks
7 Add whites

Recipe 2

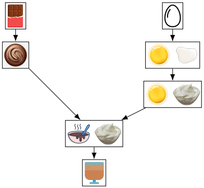
1 Get eggs
4 Separate eggs
5 Beat whites
2 Get chocolate
3 Warm chocolate
6 Add yolks
7 Add whites

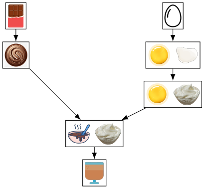
Recipe 3

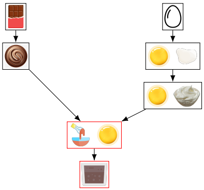
1 Get eggs
2 Get chocolate
3 Warm chocolate
4 Separate eggs
5 Beat whites
7 Add whites
6 Add yolks

Recipe 4

1 Get eggs
2 Get chocolate
3 Warm chocolate
4 Separate eggs
6 Add yolks
5 Beat whites
7 Add whites

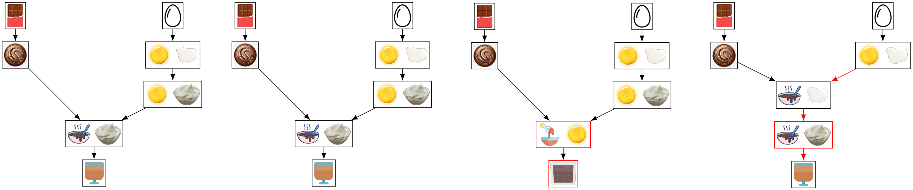






Diagrams solve the equality of recipes?

Recipe 1	Recipe 2	Recipe 3	Recipe 4
1 Get eggs	1 Get eggs	1 Get eggs	1 Get eggs
2 Get chocolate	4 Separate eggs	2 Get chocolate	2 Get chocolate
3 Warm chocolate	5 Beat whites	3 Warm chocolate	3 Warm chocolate
4 Separate eggs	2 Get chocolate	4 Separate eggs	4 Separate eggs
5 Beat whites	3 Warm chocolate	5 Beat whites	6 Add yolks
6 Add yolks	6 Add yolks	7 Add whites	5 Beat whites
7 Add whites	7 Add whites	6 Add yolks	7 Add whites

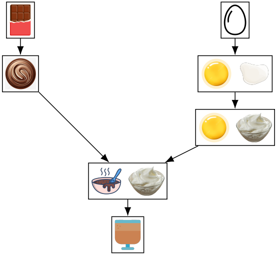


⟶ Some order in diagrams is still meaningless for causality!
Some **commutations** give different graphs but are the **same** recipe

Yet another representation of recipes

Recipe 1

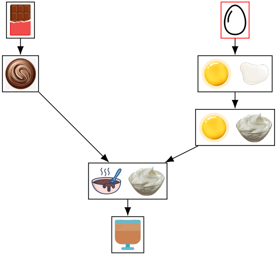
1. Get eggs
2. Get chocolate
3. Warm chocolate
4. Separate eggs
5. Beat whites
6. Add yolks
7. Add whites



Yet another representation of recipes

Recipe 1

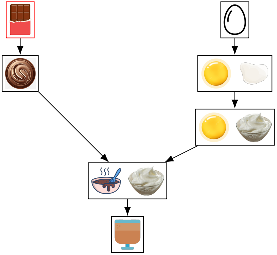
1. Get eggs
2. Get chocolate
3. Warm chocolate
4. Separate eggs
5. Beat whites
6. Add yolks
7. Add whites

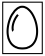


Yet another representation of recipes

Recipe 1

1. Get eggs
2. Get chocolate
3. Warm chocolate
4. Separate eggs
5. Beat whites
6. Add yolks
7. Add whites

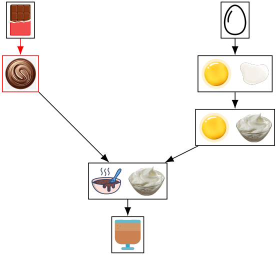




Yet another representation of recipes

Recipe 1

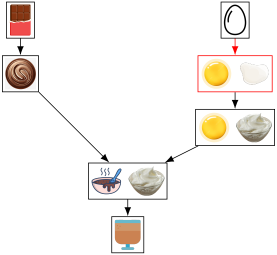
1. Get eggs
2. Get chocolate
3. Warm chocolate
4. Separate eggs
5. Beat whites
6. Add yolks
7. Add whites

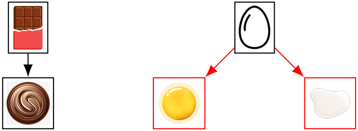


Yet another representation of recipes

Recipe 1

1. Get eggs
2. Get chocolate
3. Warm chocolate
4. Separate eggs
5. Beat whites
6. Add yolks
7. Add whites

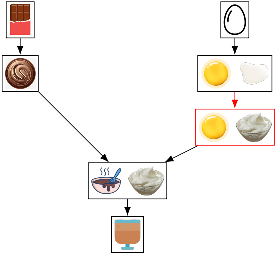


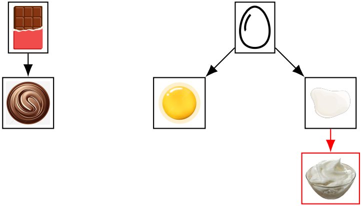


Yet another representation of recipes

Recipe 1

1. Get eggs
2. Get chocolate
3. Warm chocolate
4. Separate eggs
5. Beat whites
6. Add yolks
7. Add whites

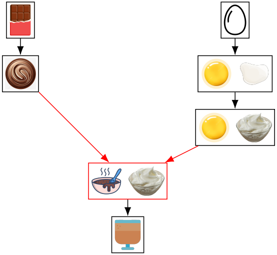


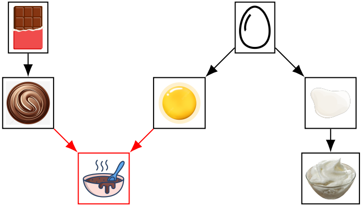


Yet another representation of recipes

Recipe 1

1. Get eggs
2. Get chocolate
3. Warm chocolate
4. Separate eggs
5. Beat whites
6. Add yolks
7. Add whites

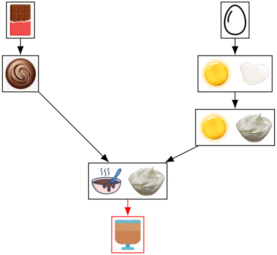


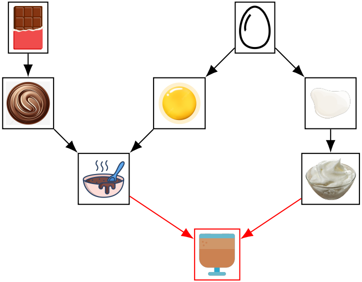


Yet another representation of recipes

Recipe 1

1. Get eggs
2. Get chocolate
3. Warm chocolate
4. Separate eggs
5. Beat whites
6. Add yolks
7. Add whites

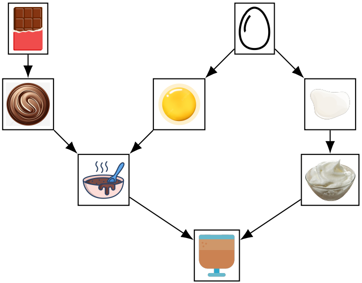


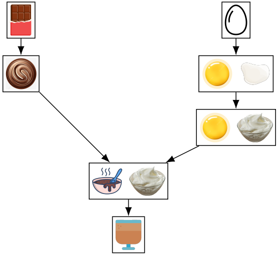


Yet another representation of recipes

Recipe 1

1 Get eggs
2 Get chocolate
3 Warm chocolate
4 Separate eggs
5 Beat whites
6 Add yolks
7 Add whites



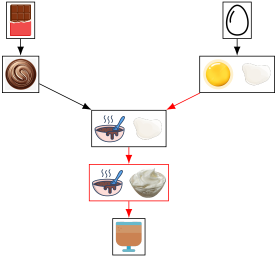


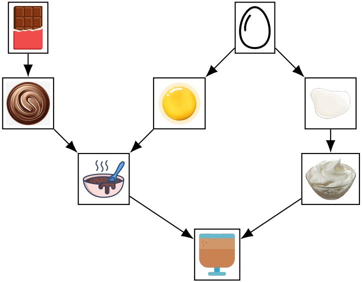
⟶ Parallelize everything
The remaining order is causality!
Unique writing for a recipe: **canonicity**

Yet another representation of recipes

Recipe 4

1. Get eggs
2. Get chocolate
3. Warm chocolate
4. Separate eggs
6. Add yolks
5. Beat whites
7. Add whites





⟶ Parallelize everything

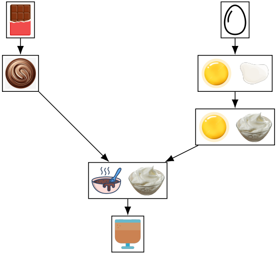
The remaining order is causality!

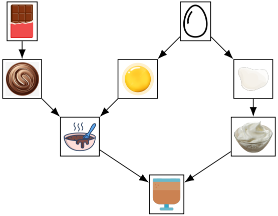
Unique writing for a recipe: **canonicity**

Three representations of recipes or proofs

Recipe 1

1. Get eggs
2. Get chocolate
3. Warm chocolate
4. Separate eggs
5. Beat whites
6. Add yolks
7. Add whites

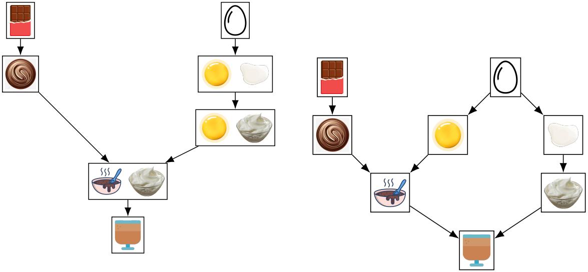




Three representations of recipes or proofs

Recipe 1

1. Get eggs
2. Get chocolate
3. Warm chocolate
4. Separate eggs
5. Beat whites
6. Add yolks
7. Add whites



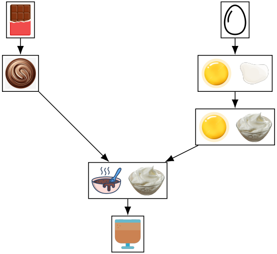
Hilbert System

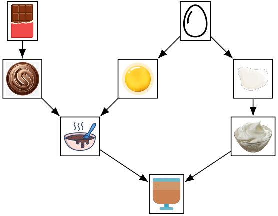
1. E
2. C
3. hC
4. Y W
5. bW
6. P
7. M

Three representations of recipes or proofs

Recipe 1

1. Get eggs
2. Get chocolate
3. Warm chocolate
4. Separate eggs
5. Beat whites
6. Add yolks
7. Add whites





Hilbert System

1. E
2. C
3. hC
4. Y W
5. bW
6. P
7. M

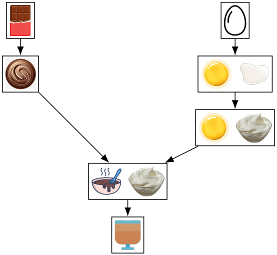
Sequent Calculus

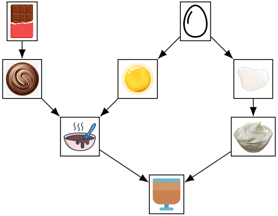
$$\dfrac{\dfrac{\dfrac{\dfrac{}{C \vdash C}}{C \vdash hC} \qquad \dfrac{\dfrac{\dfrac{}{E \vdash E}}{E \vdash Y\ W}}{E \vdash Y\ bW}}{C\ E \vdash P\ bW}}{C\ E \vdash M}$$

Three representations of recipes or proofs

Recipe 1

1. Get eggs
2. Get chocolate
3. Warm chocolate
4. Separate eggs
5. Beat whites
6. Add yolks
7. Add whites





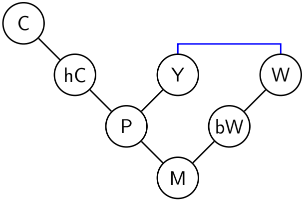
Hilbert System

1. E
2. C
3. hC
4. Y W
5. bW
6. P
7. M

Sequent Calculus

$$\dfrac{\dfrac{\dfrac{\dfrac{}{C \vdash C}}{C \vdash hC} \qquad \dfrac{\dfrac{\dfrac{}{E \vdash E}}{E \vdash Y\ W}}{E \vdash Y\ bW}}{C\ E \vdash P\ bW}}{C\ E \vdash M}$$

Proof-Net



Formulas and Connectives

Mousse Recipe

Ingredients: chocolate, eggs

$\vdots$

Recipe: produce mousse from chocolate and eggs

Formulas and Connectives

Mousse Recipe

Ingredients: chocolate, eggs

⋮

C and E

Formulas and Connectives

Mousse Recipe

Ingredients: chocolate, eggs

⋮

$C \wedge E$

$\wedge$ and, take both

Formulas and Connectives

Dark or Milk Mousse

Ingredients: dark or milk chocolate, eggs

⋮

$(dC \vee mC) \wedge E$

$\wedge$ and, take both

$\vee$ or, take one

Formulas and Connectives

Organic or Regular, Dark or Milk Mousse

Ingredients: dark or milk chocolate, organic or regular eggs

⋮

$$(dC \vee mC) \wedge (oE \vee rE)$$

$\wedge$ and, take both

$\vee$ or, take one

Formulas and Connectives

Organic or Regular, Dark or Milk Mousse

Ingredients: dark or milk chocolate, organic or regular eggs

⋮

$$(dC \vee mC) \otimes (oE \vee rE)$$

$\otimes$ and, take both

$\vee$ or, take one

Formulas and Connectives

Mousses at a Restaurant

Desserts

Dark Chocolate Mousse

or

Milk Chocolate Mousse

(organic or regular eggs according to supplies)

$$(dC \vee mC) \otimes (oE \vee rE)$$

- $\otimes$ and, take both
- $\vee$ or, take one

Formulas and Connectives

Mousses at a Restaurant

Desserts

Dark Chocolate Mousse

or

Milk Chocolate Mousse

(organic or regular eggs according to supplies)

$$(dC \mathbin{\&} mC) \otimes (oE \oplus rE)$$

- $\otimes$ and, take both
- $\&$ or, **you** choose one
- $\oplus$ or, **someone else** chooses one

Equality of Formulas

- $\otimes$ and, take both
- $\&$ or, **you** choose one
- $\oplus$ or, **someone else** chooses one

- compare $C \otimes E$ and $E \otimes C$

Equality of Formulas

$\otimes$ and, take both

$\&$ or, **you** choose one

$\oplus$ or, **someone else** chooses one

- compare $C \otimes E$ and $E \otimes C$
- compare $C \otimes (Y \otimes W)$ and $(C \otimes Y) \otimes W$

Equality of Formulas

$\otimes$ and, take both

$\&$ or, **you** choose one

$\oplus$ or, **someone else** chooses one

- compare $C \otimes E$ and $E \otimes C$
- compare $C \otimes (Y \otimes W)$ and $(C \otimes Y) \otimes W$
- compare $C \otimes (oE \oplus rE)$ and $(C \otimes oE) \oplus (C \otimes rE)$

Equality of Formulas

$\otimes$ and, take both

$\&$ or, **you** choose one

$\oplus$ or, **someone else** chooses one

- compare $C \otimes E$ and $E \otimes C$
- compare $C \otimes (Y \otimes W)$ and $(C \otimes Y) \otimes W$
- compare $C \otimes (oE \oplus rE)$ and $(C \otimes oE) \oplus (C \otimes rE)$

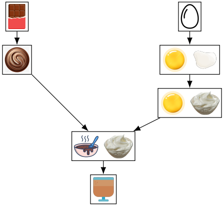
Again, **different syntaxes / writings** for a **same** underlying object

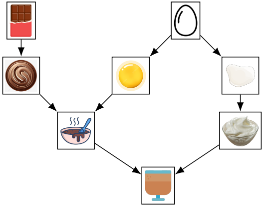
Transparent ways to go from one formula to the other, without losses
$\rightarrow$ isomorphism

$$
\begin{aligned}
C \otimes E &\simeq E \otimes C && \text{(associativity)} \\
C \otimes (Y \otimes W) &\simeq (C \otimes Y) \otimes W && \text{(commutativity)} \\
C \otimes (oE \oplus rE) &\simeq (C \otimes oE) \oplus (C \otimes rE) && \text{(distributivity)}
\end{aligned}
$$

Recipe 1

1. Get eggs
2. Get chocolate
3. Warm chocolate
4. Separate eggs
5. Beat whites
6. Add yolks
7. Add whites





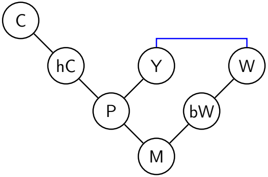
Hilbert System

1. E
2. C
3. hC
4. Y W
5. bW
6. P
7. M

Sequent Calculus

$$\dfrac{\dfrac{\dfrac{\dfrac{}{C \vdash C}}{C \vdash hC} \qquad \dfrac{\dfrac{\dfrac{}{E \vdash E}}{E \vdash Y\ W}}{E \vdash Y\ bW}}{C\ E \vdash P\ bW}}{C\ E \vdash M}$$

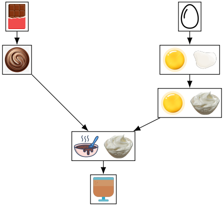
Proof-Net

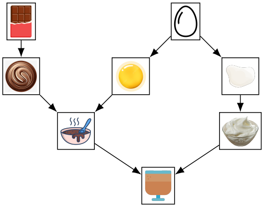


Multiplicative-Additive Linear Logic

Recipe 1

1. Get eggs
2. Get chocolate
3. Warm chocolate
4. Separate eggs
5. Beat whites
6. Add yolks
7. Add whites





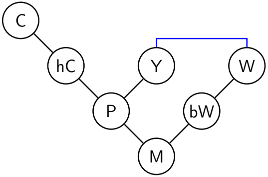
Hilbert System

1. E
2. C
3. hC
4. Y W
5. bW
6. P
7. M

Sequent Calculus

$$\dfrac{\dfrac{\dfrac{\dfrac{}{C \vdash C}}{C \vdash hC} \qquad \dfrac{\dfrac{\dfrac{}{E \vdash E}}{E \vdash Y\ W}}{E \vdash Y\ bW}}{C\ E \vdash P\ bW}}{C\ E \vdash M}$$

Proof-Net



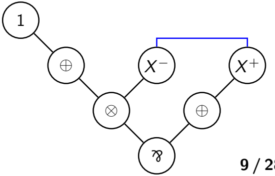
Hilbert System

1. X^-, X^+
2. 1
3. $1 \oplus H$
4. $X^-, B \oplus X^+$
5. $((1 \oplus H) \otimes X^-), B \oplus X^+$
6. $((1 \oplus H) \otimes X^-) \parr (B \oplus X^+)$

Sequent Calculus

$$\dfrac{\dfrac{\dfrac{\dfrac{}{\vdash 1}}{\vdash 1 \oplus H} \qquad \dfrac{\dfrac{}{\vdash X^-, X^+}}{\vdash X^-, B \oplus X^+}}{\vdash (1 \oplus H) \otimes X^-, B \oplus X^+}}{\vdash ((1 \oplus H) \otimes X^-) \parr (B \oplus X^+)}$$

Proof-Net



Multiplicative-Additive Linear Logic

$$A, B ::= \overbrace{X^+ \mid X^-}^{\textit{atoms}} \mid \overbrace{A \underset{\textit{and}}{\otimes} B \mid A \underset{\textit{or}}{\parr} B \mid \underset{\textit{true}}{1} \mid \underset{\textit{false}}{\bot}}^{\textit{multiplicative}}$$

$$\mid \underbrace{A \overset{\textit{and}}{\&} B \mid A \overset{\textit{or}}{\oplus} B \mid \overset{\textit{true}}{\top} \mid \overset{\textit{false}}{0}}_{\textit{additive}}$$

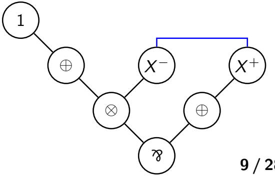
Hilbert System

1. X^-, X^+
2. 1
3. $1 \oplus H$
4. $X^-, B \oplus X^+$
5. $((1 \oplus H) \otimes X^-), B \oplus X^+$
6. $((1 \oplus H) \otimes X^-) \parr (B \oplus X^+)$

Sequent Calculus

$$\dfrac{\dfrac{\dfrac{\dfrac{}{\vdash 1}}{\vdash 1 \oplus H} \qquad \dfrac{\dfrac{}{\vdash X^-, X^+}}{\vdash X^-, B \oplus X^+}}{\vdash (1 \oplus H) \otimes X^-, B \oplus X^+}}{\vdash ((1 \oplus H) \otimes X^-) \parr (B \oplus X^+)}$$

Proof-Net



Multiplicative-Additive Linear Logic

$$A, B \;::=\; \overbrace{X^{+} \mid X^{-}}^{\textit{atoms}} \mid \overbrace{A \underset{\textit{and}}{\otimes} B \mid A \underset{\textit{or}}{⅋} B \mid \underset{\textit{true}}{1} \mid \underset{\textit{false}}{\bot}}^{\textit{multiplicative}}$$

$$\mid \underbrace{A \overset{\textit{and}}{\&} B \mid A \overset{\textit{or}}{\oplus} B \mid \overset{\textit{true}}{\top} \mid \overset{\textit{false}}{0}}_{\textit{additive}}$$

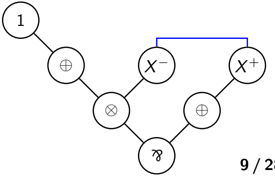
Hilbert System

1. X^{-}, X^{+}
2. 1
3. $1 \oplus H$
4. $X^{-}, B \oplus X^{+}$
5. $((1 \oplus H) \otimes X^{-}), B \oplus X^{+}$
6. $((1 \oplus H) \otimes X^{-}) ⅋ (B \oplus X^{+})$

Sequent Calculus

$$\dfrac{\dfrac{\dfrac{}{\vdash 1}}{\vdash 1 \oplus H} \qquad \dfrac{\dfrac{}{\vdash X^{-}, X^{+}}}{\vdash X^{-}, B \oplus X^{+}}}{\dfrac{\vdash (1 \oplus H) \otimes X^{-}, B \oplus X^{+}}{\vdash ((1 \oplus H) \otimes X^{-}) ⅋ (B \oplus X^{+})}}$$

Proof-Net



Plan

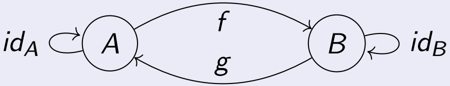
- Isomorphisms in Multiplicative-Additive Linear Logic

- Retractions in Multiplicative Linear Logic

Isomorphisms

$C \otimes E$ behaves the same as $E \otimes C$ $\rightarrow$ isomorphism $C \otimes E \simeq E \otimes C$

In category theory: isomorphism $A \simeq B$

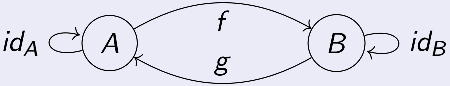


$$g \circ f = id_A$$
$$f \circ g = id_B$$

Isomorphisms

$C \otimes E$ behaves the same as $E \otimes C$ $\rightarrow$ isomorphism $C \otimes E \simeq E \otimes C$

In category theory: isomorphism $A \simeq B$



$$g \circ f = id_A$$
$$f \circ g = id_B$$

In λ-calculus: isomorphism $A \simeq B$

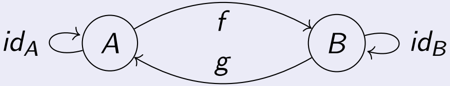
Terms $M : A \rightarrow B$ and $N : B \rightarrow A$ such that

$$\lambda x^A.N(Mx) =_{\beta\eta} \lambda x^A.x \qquad \text{and} \qquad \lambda y^B.M(Ny) =_{\beta\eta} \lambda y^B.y$$

Isomorphisms

$C \otimes E$ behaves the same as $E \otimes C$ $\rightarrow$ isomorphism $C \otimes E \simeq E \otimes C$

In category theory: isomorphism $A \simeq B$



$$g \circ f = id_A$$
$$f \circ g = id_B$$

In λ-calculus: isomorphism $A \simeq B$

Terms $M : A \rightarrow B$ and $N : B \rightarrow A$ such that

$$\lambda x^A.N(Mx) =_{\beta\eta} \lambda x^A.x \qquad \text{and} \qquad \lambda y^B.M(Ny) =_{\beta\eta} \lambda y^B.y$$

In (linear) logic: isomorphism $A \simeq B$

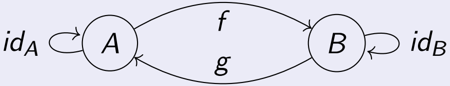
Proofs π of $A \vdash B$ and ϕ of $B \vdash A$ such that

$$\dfrac{\overset{\pi}{A \vdash B} \qquad \overset{\phi}{B \vdash A}}{A \vdash A}\,(cut) =_{\beta\eta} \dfrac{}{A \vdash A}\,(ax) \quad \text{and} \quad \dfrac{\overset{\phi}{B \vdash A} \qquad \overset{\pi}{A \vdash B}}{B \vdash B}\,(cut) =_{\beta\eta} \dfrac{}{B \vdash B}\,(ax)$$

Isomorphisms

$C \otimes E$ behaves the same as $E \otimes C$ $\to$ isomorphism $C \otimes E \simeq E \otimes C$

In category theory: isomorphism $A \simeq B$



$$g \circ f = id_A$$
$$f \circ g = id_B$$

In λ-calculus: isomorphism $A \simeq B$

Terms $M : A \to B$ and $N : B \to A$ such that

$$\lambda x^A.N(Mx) =_{\beta\eta} \lambda x^A.x \qquad \text{and} \qquad \lambda y^B.M(Ny) =_{\beta\eta} \lambda y^B.y$$

In (linear) logic: isomorphism $A \simeq B$

Proofs π of $A \vdash B$ and ϕ of $B \vdash A$ such that

$$\frac{\overset{\pi}{A \vdash B} \quad \overset{\phi}{B \vdash A}}{A \vdash A}\ (cut) =_{\beta\eta} \frac{}{A \vdash A}\ (ax) \quad \text{and} \quad \frac{\overset{\phi}{B \vdash A} \quad \overset{\pi}{A \vdash B}}{B \vdash B}\ (cut) =_{\beta\eta} \frac{}{B \vdash B}\ (ax)$$

rule commutations $\overset{r}{\vdash\!\dashv} \subseteq =_{\beta\eta}$

Literature on isomorphisms

Goal: obtain an equational theory

Syntactic Method

Analyze pairs of proofs of isos
$\rightarrow$ *get information on their formulas*

Semantic Method

Find a model with the same isos but where computation/equality is easy

Literature on isomorphisms

Goal: obtain an equational theory

Syntactic Method	Semantic Method
Analyze pairs of proofs of isos $\rightarrow$ *get information on their formulas*	*Find a model with the same isos but where computation/equality is easy*

λ-calculus with products and unit type
Cartesian closed categories

Semantic (finite sets) [Sol83]

$A \times (B \times C) \simeq (A \times B) \times C$	$A \times B \simeq B \times A$	$1 \times A \simeq A$
$(A \times B) \rightarrow C \simeq A \rightarrow (B \rightarrow C)$		$1 \rightarrow A \simeq A$
$A \rightarrow (B \times C) \simeq (A \rightarrow B) \times (A \rightarrow C)$		$A \rightarrow 1 \simeq 1$

Reduces to Tarski's High School Algebra Problem:
can all equalities involving product, exponential and 1 be found using only

$$a(bc) = (ab)c \quad ab = ba \quad 1a = a$$
$$c^{ab} = (c^b)^a \qquad a^1 = a$$
$$(bc)^a = b^a c^a \qquad 1^a = 1$$

Literature on isomorphisms

Goal: obtain an equational theory

Syntactic Method

Analyze pairs of proofs of isos
$\rightarrow$ get information on their formulas

Semantic Method

Find a model with the same isos but where computation/equality is easy

$\begin{cases}\text{Multiplicative Linear Logic}\\ \star\text{-autonomous categories}\end{cases}$ Syntactic (proof-nets) [BD99]

Associativity	$A \otimes (B \otimes C) \simeq (A \otimes B) \otimes C$ $A \parr (B \parr C) \simeq (A \parr B) \parr C$
Commutativity	$A \otimes B \simeq B \otimes A \quad A \parr B \simeq B \parr A$
Neutrality	$A \otimes 1 \simeq A \quad A \parr \bot \simeq A$

Literature on isomorphisms

Goal: obtain an equational theory

Syntactic Method

Analyze pairs of proofs of isos
$\rightarrow$ get information on their formulas

Semantic Method

Find a model with the same isos but where computation/equality is easy

$\begin{cases}\text{Polarized Linear Logic} \\ \text{Control Categories}\end{cases}$ Semantic (games, forest isomorphisms) [Lau05]

Associativity	$A \otimes (B \otimes C) \simeq (A \otimes B) \otimes C$ $A ⅋ (B ⅋ C) \simeq (A ⅋ B) ⅋ C$		$A \oplus (B \oplus C) \simeq (A \oplus B) \oplus C$ $A \& (B \& C) \simeq (A \& B) \& C$	
Commutativity	$A \otimes B \simeq B \otimes A$	$A ⅋ B \simeq B ⅋ A$	$A \oplus B \simeq B \oplus A$	$A \& B \simeq B \& A$
Neutrality	$A \otimes 1 \simeq A$	$A ⅋ \bot \simeq A$	$A \oplus 0 \simeq A$	$A \& \top \simeq A$
Distributivity	$A \otimes (B \oplus C) \simeq (A \otimes B) \oplus (A \otimes C)$		$A ⅋ (B \& C) \simeq (A ⅋ B) \& (A ⅋ C)$	
Annihilation	$A \otimes 0 \simeq 0$		$A ⅋ \top \simeq \top$	
Seely	$!(A \& B) \simeq !A \otimes !B$ $!\top \simeq 1$		$?(A \oplus B) \simeq ?A ⅋ ?B$ $?0 \simeq \bot$	

Literature on isomorphisms

<u>Goal:</u> obtain an equational theory

Syntactic Method

Analyze pairs of proofs of isos
→ get information on their formulas

Semantic Method

Find a model with the same isos but where computation/equality is easy

{ Polarized Linear Logic
{ Control Categories

Semantic (games, forest isomorphisms) [Lau05]

Associativity	$A \otimes (B \otimes C) \simeq (A \otimes B) \otimes C$ $A \parr (B \parr C) \simeq (A \parr B) \parr C$		$A \oplus (B \oplus C) \simeq (A \oplus B) \oplus C$ $A \,\&\, (B \,\&\, C) \simeq (A \,\&\, B) \,\&\, C$	
Commutativity	$A \otimes B \simeq B \otimes A$	$A \parr B \simeq B \parr A$	$A \oplus B \simeq B \oplus A$	$A \,\&\, B \simeq B \,\&\, A$
Neutrality	$A \otimes 1 \simeq A$	$A \parr \bot \simeq A$	$A \oplus 0 \simeq A$	$A \,\&\, \top \simeq A$
Distributivity	$A \otimes (B \oplus C) \simeq (A \otimes B) \oplus (A \otimes C)$		$A \parr (B \,\&\, C) \simeq (A \parr B) \,\&\, (A \parr C)$	
Annihilation	$A \otimes 0 \simeq 0$		$A \parr \top \simeq \top$	
Seely	$!(A \,\&\, B) \simeq\, !A \otimes !B$ $!\top \simeq 1$		$?(A \oplus B) \simeq\, ?A \parr ?B$ $?0 \simeq \bot$	

~~$\top \otimes A$~~

Literature on isomorphisms

Goal: obtain an equational theory

Syntactic Method

Analyze pairs of proofs of isos
→ get information on their formulas

Semantic Method

Find a model with the same isos but where computation/equality is easy

{ Multiplicative-Additive Linear Logic
{ ⋆-autonomous categories with finite products

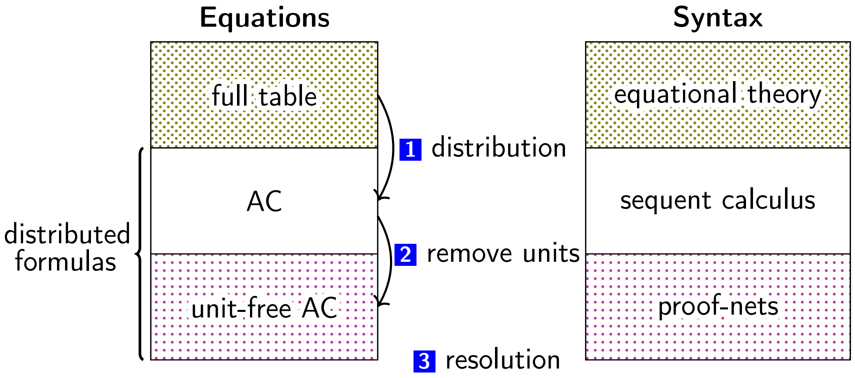
Syntactic [thesis]

Associativity	$A \otimes (B \otimes C) \simeq (A \otimes B) \otimes C$ $A \parr (B \parr C) \simeq (A \parr B) \parr C$		$A \oplus (B \oplus C) \simeq (A \oplus B) \oplus C$ $A \,\&\, (B \,\&\, C) \simeq (A \,\&\, B) \,\&\, C$	
Commutativity	$A \otimes B \simeq B \otimes A$	$A \parr B \simeq B \parr A$	$A \oplus B \simeq B \oplus A$	$A \,\&\, B \simeq B \,\&\, A$
Neutrality	$A \otimes 1 \simeq A$	$A \parr \bot \simeq A$	$A \oplus 0 \simeq A$	$A \,\&\, \top \simeq A$
Distributivity	$A \otimes (B \oplus C) \simeq (A \otimes B) \oplus (A \otimes C)$		$A \parr (B \,\&\, C) \simeq (A \parr B) \,\&\, (A \parr C)$	
Annihilation	$A \otimes 0 \simeq 0$		$A \parr \top \simeq \top$	
Seely	$!(A \,\&\, B) \simeq !A \otimes !B$ $!\top \simeq 1$		$?(A \oplus B) \simeq ?A \parr ?B$ $?0 \simeq \bot$	

Proof sketch

Syntactic method:

1. Simplify using the **distributivity** equations *(rewriting theory)*
2. Remove the units *(sequent calculus)*
3. Analyze the shape of isomorphisms to conclude *(proof-nets)*



Proof 1/3: Distribution

Distributed Formula

Associativity	$A \otimes (B \otimes C) \simeq (A \otimes B) \otimes C$ $A \oplus (B \oplus C) \simeq (A \oplus B) \oplus C$		$A ⅋ (B ⅋ C) \simeq (A ⅋ B) ⅋ C$ $A \& (B \& C) \simeq (A \& B) \& C$	
Commutativity	$A \otimes B \simeq B \otimes A$	$A ⅋ B \simeq B ⅋ A$	$A \oplus B \simeq B \oplus A$	$A \& B \simeq B \& A$
Neutrality	$A \otimes 1 \rightarrow A$	$A ⅋ \bot \rightarrow A$	$A \oplus 0 \rightarrow A$	$A \& \top \rightarrow A$
Distributivity	$A \otimes (B \oplus C) \rightarrow (A \otimes B) \oplus (A \otimes C)$		$A ⅋ (B \& C) \rightarrow (A ⅋ B) \& (A ⅋ C)$	
Annihilation	$A \otimes 0 \rightarrow 0$		$A ⅋ \top \rightarrow \top$	

Proof 1/3: Distribution

Distributed Formula

Associativity	$A \otimes (B \otimes C) \simeq (A \otimes B) \otimes C$ $A \oplus (B \oplus C) \simeq (A \oplus B) \oplus C$		$A \parr (B \parr C) \simeq (A \parr B) \parr C$ $A \& (B \& C) \simeq (A \& B) \& C$	
Commutativity	$A \otimes B \simeq B \otimes A$	$A \parr B \simeq B \parr A$	$A \oplus B \simeq B \oplus A$	$A \& B \simeq B \& A$
Neutrality	$A \otimes 1 \to A$	$A \parr \bot \to A$	$A \oplus 0 \to A$	$A \& \top \to A$
Distributivity	$A \otimes (B \oplus C) \to (A \otimes B) \oplus (A \otimes C)$		$A \parr (B \& C) \to (A \parr B) \& (A \parr C)$	
Annihilation	$A \otimes 0 \to 0$		$A \parr \top \to \top$	

Proposition

$\mathcal{E}$ *complete for* ***distributed*** *formulas*

$\Downarrow$

$\mathcal{E}$ + *Neut.* + *Dist.* + *Anni.* *complete for* ***all*** *formulas*

Associativity	$A \otimes (B \otimes C) \simeq (A \otimes B) \otimes C$ $A \oplus (B \oplus C) \simeq (A \oplus B) \oplus C$		$A \parr (B \parr C) \simeq (A \parr B) \parr C$ $A \& (B \& C) \simeq (A \& B) \& C$	
Commutativity	$A \otimes B \simeq B \otimes A$	$A \parr B \simeq B \parr A$	$A \oplus B \simeq B \oplus A$	$A \& B \simeq B \& A$
Neutrality	$A \otimes 1 \simeq A$	$A \parr \bot \simeq A$	$A \oplus 0 \simeq A$	$A \& \top \simeq A$
Distributivity	$A \otimes (B \oplus C) \simeq (A \otimes B) \oplus (A \otimes C)$		$A \parr (B \& C) \simeq (A \parr B) \& (A \parr C)$	
Annihilation	$A \otimes 0 \simeq 0$		$A \parr \top \simeq \top$	

Proof 2/3: Remove the units

In isomorphisms of *distributed* formulas: units = fresh atoms

1. In the identity: $\dfrac{}{\vdash \top, 0}\ (\top)$

$$\dfrac{\dfrac{}{\vdash 1}\ (1)}{\vdash \bot, 1}\ (\bot)$$

Proof 2/3: Remove the units

In isomorphisms of *distributed* formulas: units = fresh atoms

1. In the identity:

$$\frac{}{\vdash \top, 0}\,(\top) \quad \longrightarrow \quad \frac{}{\vdash X^-, X^+}\,(ax)$$

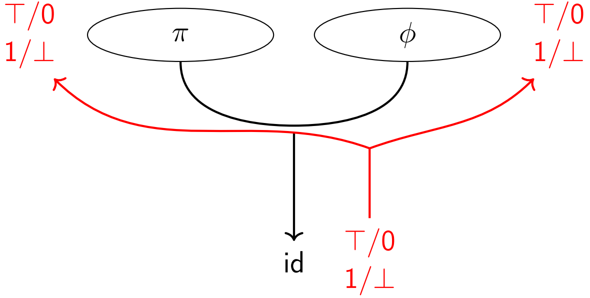
$$\frac{\frac{}{\vdash 1}\,(1)}{\vdash \bot, 1}\,(\bot) \quad \longrightarrow \quad \frac{}{\vdash Y^-, Y^+}\,(ax)$$

Proof 2/3: Remove the units

In isomorphisms of *distributed* formulas: units = fresh atoms

1. In the identity: $\dfrac{}{\vdash \top, 0}\ (\top) \quad \longrightarrow \quad \dfrac{}{\vdash X^-, X^+}\ (ax)$

$$\dfrac{\dfrac{}{\vdash 1}\ (1)}{\vdash \bot, 1}\ (\bot) \quad \longrightarrow \quad \dfrac{}{\vdash Y^-, Y^+}\ (ax)$$



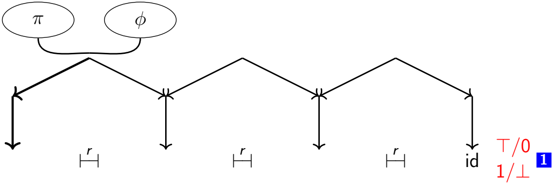
Proof 2/3: Remove the units

In isomorphisms of *distributed* formulas: units = fresh atoms

1. In the identity:

$$\frac{}{\vdash \top, 0}\,(\top) \quad \longrightarrow \quad \frac{}{\vdash X^-, X^+}\,(ax)$$

$$\frac{\frac{}{\vdash 1}\,(1)}{\vdash \bot, 1}\,(\bot) \quad \longrightarrow \quad \frac{}{\vdash Y^-, Y^+}\,(ax)$$



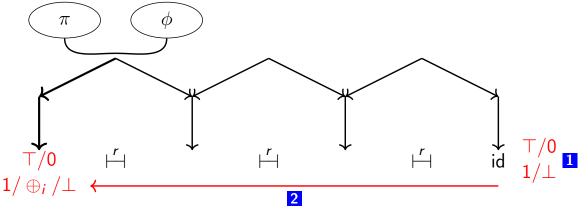
Proof 2/3: Remove the units

In isomorphisms of *distributed* formulas: units = fresh atoms

1. In the identity: $\dfrac{}{\vdash \top, 0}\ (\top) \quad \longrightarrow \quad \dfrac{}{\vdash X^-, X^+}\ (ax)$

   $\dfrac{\dfrac{}{\vdash 1}\ (1)}{\vdash \bot, 1}\ (\bot) \quad \longrightarrow \quad \dfrac{}{\vdash Y^-, Y^+}\ (ax)$

2. Shape preserved by $\overset{r}{\vdash\!\dashv}$: $\dfrac{\dfrac{\dfrac{}{\vdash 1}\ (1)}{\vdash F}\ \oplus_i}{\vdash \bot, F}\ (\bot)$ using *distributivity*



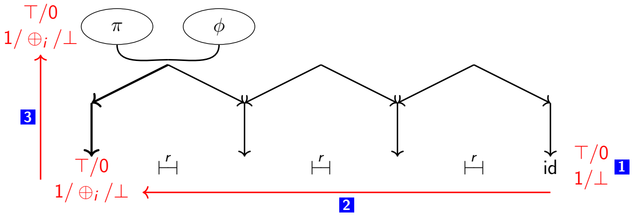
Proof 2/3: Remove the units

In isomorphisms of *distributed* formulas: units = fresh atoms

1. In the identity: $\dfrac{}{\vdash \top, 0}\,(\top) \quad \longrightarrow \quad \dfrac{}{\vdash X^-, X^+}\,(ax)$

   $$\dfrac{\dfrac{}{\vdash 1}\,(1)}{\vdash \bot, 1}\,(\bot) \quad \longrightarrow \quad \dfrac{}{\vdash Y^-, Y^+}\,(ax)$$

2. Shape preserved by $\overset{r}{\vdash\!\dashv}$: $\dfrac{\dfrac{\dfrac{}{\vdash 1}\,(1)}{\vdash F}\,\oplus_i}{\vdash \bot, F}\,(\bot)$ using *distributivity*
3. Cut-elimination in isomorphisms cannot "completely" erase units rules



Proof 2/3: Remove the units

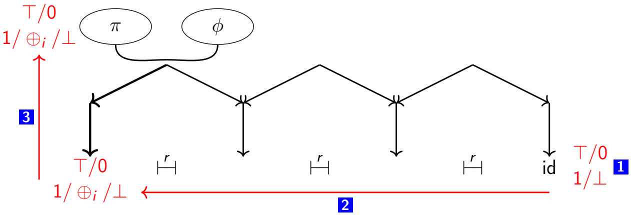
In isomorphisms of *distributed* formulas: units = fresh atoms

1. In the identity: $\dfrac{}{\vdash \top, 0}\,(\top) \longrightarrow \dfrac{}{\vdash X^-, X^+}\,(ax)$

   $\dfrac{\dfrac{}{\vdash 1}\,(1)}{\vdash \bot, 1}\,(\bot) \longrightarrow \dfrac{}{\vdash Y^-, Y^+}\,(ax)$

2. Shape preserved by $\overset{r}{\longmapsto}$: $\dfrac{\dfrac{\dfrac{}{\vdash 1}\,(1)}{\vdash F}\,\oplus_i}{\vdash \bot, F}\,(\bot)$ using *distributivity*

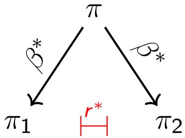
3. Cut-elimination in isomorphisms cannot "completely" erase units rules



⟹ No more units!

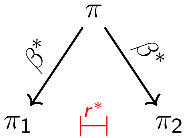
Parenthesis: Confluence up to

Needed:

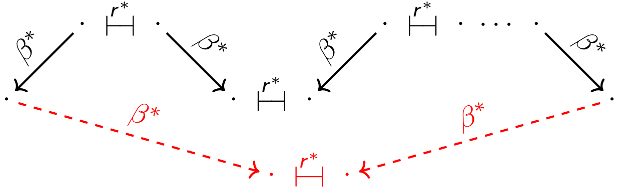


Parenthesis: Confluence up to

Needed:

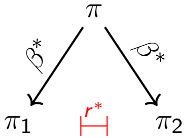


Generalization: Church-Rosser modulo

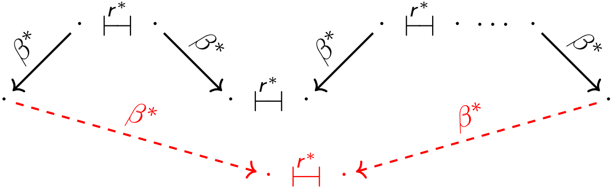


Parenthesis: Confluence up to

Needed:



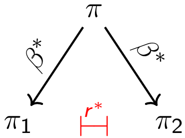
Generalization: Church-Rosser modulo



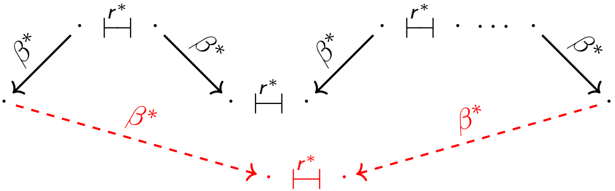
$$\Longrightarrow \quad MALL/=_{\beta\eta} = MALL/\overset{r}{\longmapsto} = \text{proof-nets}$$

Parenthesis: Confluence up to

Needed:



Generalization: Church-Rosser modulo

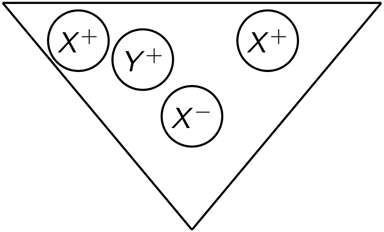


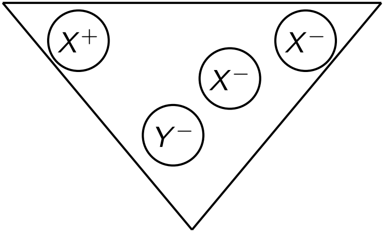
$$\Longrightarrow \quad MALL/ =_{\beta\eta} = MALL/ \overset{r}{\vdash\!\dashv} \;=\; \text{proof-nets}$$

- Already proved for MALL [CP05]
- We reproved it by showing **Strong Normalization** and using a theorem from rewriting theory [Hue80]

Proof 3/3: Shape of distributed isomorphisms

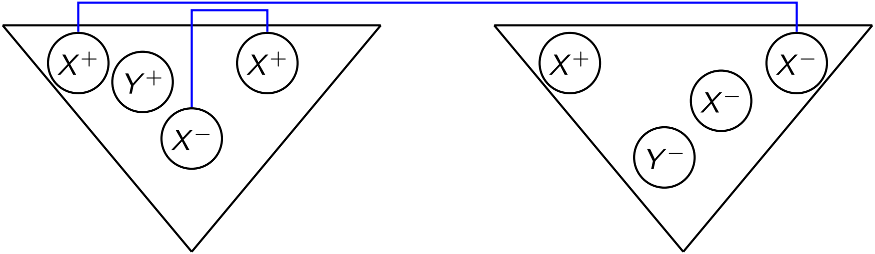
Use **proof-nets**





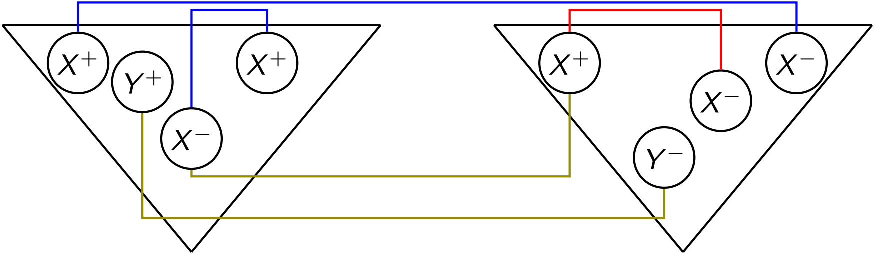
Proof 3/3: Shape of distributed isomorphisms

Use **proof-nets**



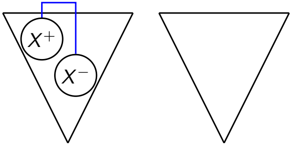
Proof 3/3: Shape of distributed isomorphisms

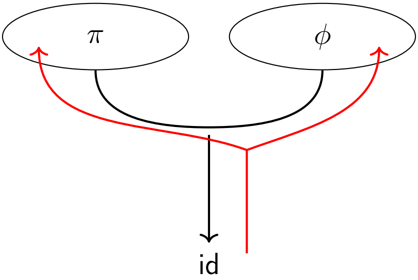
Use **proof-nets**



Proof 3/3: Shape of distributed isomorphisms

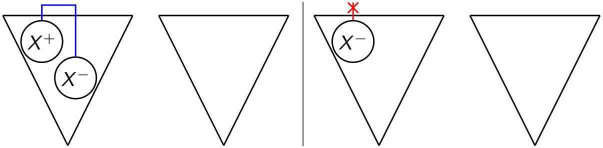
Forbidden configurations in distributed isomorphisms:

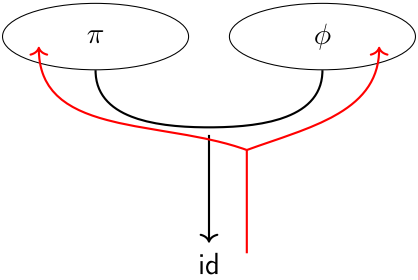




Proof 3/3: Shape of distributed isomorphisms

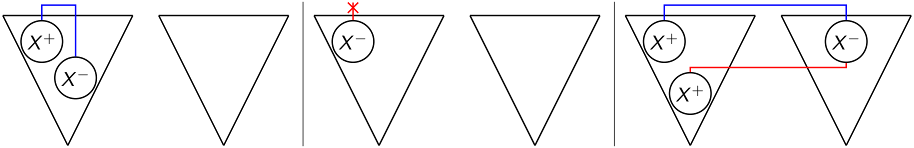
Forbidden configurations in distributed isomorphisms:

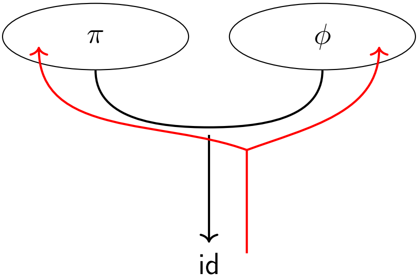




Proof 3/3: Shape of distributed isomorphisms

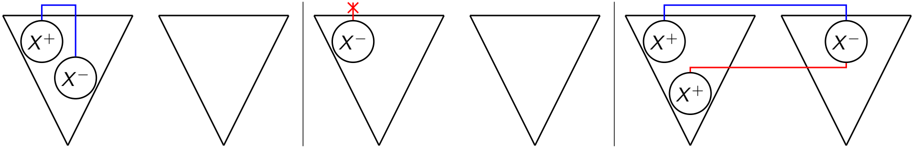
Forbidden configurations in distributed isomorphisms:



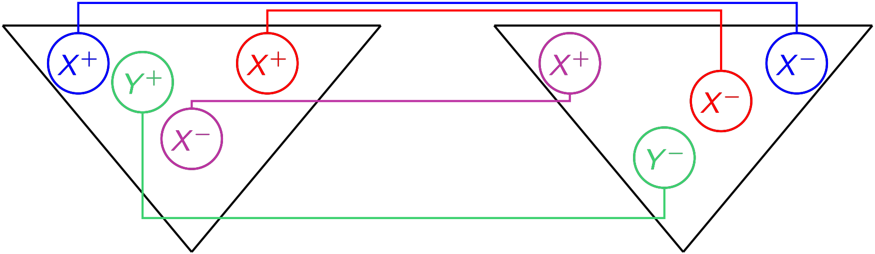


Proof 3/3: Shape of distributed isomorphisms

Forbidden configurations in distributed isomorphisms:



General shape:



$\longrightarrow$ only reordering = AC!

Plan

- **Isomorphisms in Multiplicative-Additive Linear Logic**

Associativity	$A \otimes (B \otimes C) \simeq (A \otimes B) \otimes C$ $A \parr (B \parr C) \simeq (A \parr B) \parr C$		$A \oplus (B \oplus C) \simeq (A \oplus B) \oplus C$ $A \& (B \& C) \simeq (A \& B) \& C$	
Commutativity	$A \otimes B \simeq B \otimes A$	$A \parr B \simeq B \parr A$	$A \oplus B \simeq B \oplus A$	$A \& B \simeq B \& A$
Neutrality	$A \otimes 1 \simeq A$	$A \parr \bot \simeq A$	$A \oplus 0 \simeq A$	$A \& \top \simeq A$
Distributivity	$A \otimes (B \oplus C) \simeq (A \otimes B) \oplus (A \otimes C)$		$A \parr (B \& C) \simeq (A \parr B) \& (A \parr C)$	
Annihilation	$A \otimes 0 \simeq 0$		$A \parr \top \simeq \top$	

- Retractions in Multiplicative Linear Logic

Plan

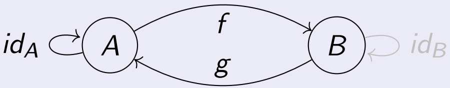
- Isomorphisms in Multiplicative-Additive Linear Logic

Associativity	$A \otimes (B \otimes C) \simeq (A \otimes B) \otimes C$ $A ⅋ (B ⅋ C) \simeq (A ⅋ B) ⅋ C$		$A \oplus (B \oplus C) \simeq (A \oplus B) \oplus C$ $A \& (B \& C) \simeq (A \& B) \& C$	
Commutativity	$A \otimes B \simeq B \otimes A$	$A ⅋ B \simeq B ⅋ A$	$A \oplus B \simeq B \oplus A$	$A \& B \simeq B \& A$
Neutrality	$A \otimes 1 \simeq A$	$A ⅋ \bot \simeq A$	$A \oplus 0 \simeq A$	$A \& \top \simeq A$
Distributivity	$A \otimes (B \oplus C) \simeq (A \otimes B) \oplus (A \otimes C)$		$A ⅋ (B \& C) \simeq (A ⅋ B) \& (A ⅋ C)$	
Annihilation	$A \otimes 0 \simeq 0$		$A ⅋ \top \simeq \top$	

- **Retractions in Multiplicative Linear Logic**

Retractions

In category theory: retraction $A \trianglelefteq B$

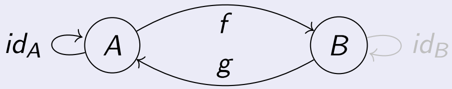


$g \circ f = id_A$

$f \circ g = id_B$

Retractions

In category theory: retraction $A \trianglelefteq B$



$g \circ f = id_A$

$f \circ g = id_B$

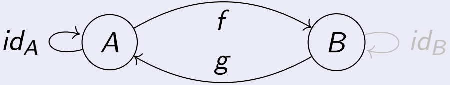
$\longrightarrow$ Natural notion of sub-typing

Example

bool $\trianglelefteq$ nat with $f(b) := \begin{cases} 0 & \text{if } b \\ 1 & \text{otherwise} \end{cases}$ and $g(n) := \begin{cases} \mathit{true} & \text{if } n \geq 1 \\ \mathit{false} & \text{otherwise} \end{cases}$

Retractions

In category theory: retraction $A \trianglelefteq B$



$$g \circ f = id_A$$
$$f \circ g = id_B$$

$\longrightarrow$ Natural notion of sub-typing

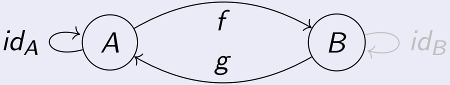
In (linear) logic: retraction $A \trianglelefteq B$

Proofs π of $A \vdash B$ and ϕ of $B \vdash A$ such that

$$\dfrac{\overset{\pi}{A \vdash B} \qquad \overset{\phi}{B \vdash A}}{A \vdash A}\,(cut) =_{\beta\eta} \dfrac{}{A \vdash A}\,(ax) \quad \text{and} \quad \dfrac{\overset{\phi}{B \vdash A} \qquad \overset{\pi}{A \vdash B}}{B \vdash B}\,(cut) =_{\beta\eta} \dfrac{}{B \vdash B}\,(ax)$$

Retractions

In category theory: retraction $A \trianglelefteq B$



$$g \circ f = id_A$$
$$f \circ g = id_B$$

$\longrightarrow$ Natural notion of sub-typing

In (linear) logic: retraction $A \trianglelefteq B$

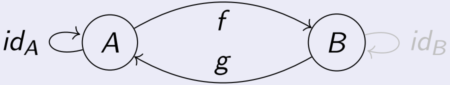
Proofs π of $A \vdash B$ and ϕ of $B \vdash A$ such that

$$\dfrac{\overset{\pi}{A \vdash B} \quad \overset{\phi}{B \vdash A}}{A \vdash A}\,(cut) =_{\beta\eta} \dfrac{}{A \vdash A}\,(ax) \quad \text{and} \quad \dfrac{\overset{\phi}{B \vdash A} \quad \overset{\pi}{A \vdash B}}{B \vdash B}\,(cut) =_{\beta\eta} \dfrac{}{B \vdash B}\,(ax)$$

$$A \trianglelefteq B \iff A^{\perp} \trianglelefteq B^{\perp}$$

Retractions

In category theory: retraction $A \trianglelefteq B$



$$g \circ f = id_A$$
$$f \circ g = id_B$$

$\longrightarrow$ Natural notion of sub-typing

In (linear) logic: retraction $A \trianglelefteq B$

Proofs π of $A \vdash B$ and ϕ of $B \vdash A$ such that

$$\frac{\overset{\pi}{A \vdash B} \quad \overset{\phi}{B \vdash A}}{A \vdash A}\,(cut) =_{\beta\eta} \frac{}{A \vdash A}\,(ax) \quad \text{and} \quad \frac{\overset{\phi}{B \vdash A} \quad \overset{\pi}{A \vdash B}}{B \vdash B}\,(cut) =_{\beta\eta} \frac{}{B \vdash B}\,(ax)$$

$$A \trianglelefteq B \iff A^{\perp} \trianglelefteq B^{\perp}$$

Example: Beffara's retraction

$$A \trianglelefteq A \otimes (A^{\perp} \parr A) \qquad \text{also} \qquad \underset{f\ b}{\overset{a}{A}} \ \underset{\longleftarrow}{\overset{\longrightarrow}{\trianglelefteq}} \ A \otimes \underset{(b,f)}{\overset{(a,id)}{(A \multimap A)}}$$

Simplifications in MLL

Syntactic method:

1. Simplify using the **neutrality** equations *(rewriting theory)*
2. Remove the units *(sequent calculus)*
3. Analyze the shape of retractions to conclude *(proof-nets)*

Simplifications in MLL

Syntactic method:

1. Simplify using the **neutrality** equations *(rewriting theory)*
2. Remove the units *(sequent calculus)*
3. Analyze the shape of retractions to conclude *(proof-nets)*

No conjecture; which *shapes* to look for?

Simplifications in MLL

Syntactic method:

1. Simplify using the **neutrality** equations *(rewriting theory)*
2. Remove the units *(sequent calculus)*
3. Analyze the shape of retractions to conclude *(proof-nets)*

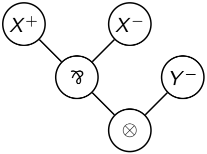
No conjecture; which *shapes* to look for?

$\longrightarrow$ Problem purely in MLL **proof-nets**!

Proof-nets for MLL

- **Formula**

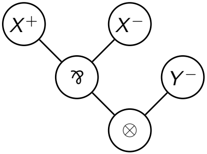
$$A, B \ ::= \ \mid X^+ \mid X^- \\ \mid A \otimes B \mid A \parr B$$



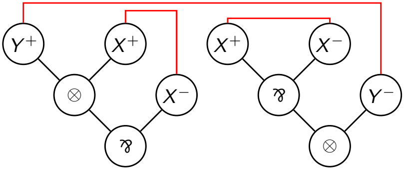
Proof-nets for MLL

- **Formula**

$$A, B \;::=\; \mid X^+ \mid X^- \mid A \otimes B \mid A \parr B$$



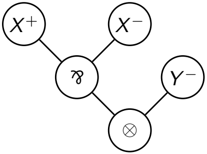
- **Proof Structure**
  formulas +
  axioms partitionning atoms



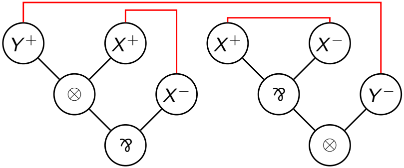
Proof-nets for MLL

- **Formula**

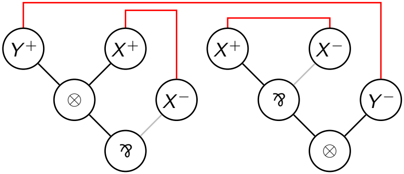
$$A, B \;::=\; \mid X^+ \mid X^- \\ \mid A \otimes B \mid A \parr B$$



- **Proof Structure**
  formulas +
  axioms partitionning atoms



- **Danos-Regnier Criterion**
  <u>acyclic</u> and <u>connected</u>
  correctness graphs



Retractions in proof-nets

In MLL Proof-Nets: retraction $A \trianglelefteq B$

Proof-nets $\mathcal{R}$ of $\vdash A^{\perp}, B$ and $\mathcal{S}$ of $\vdash B^{\perp}, A$ such that
$\mathcal{R}$ cut with $\mathcal{S}$ reduces to id

Retractions in proof-nets

In MLL Proof-Nets: retraction $A \trianglelefteq B$

Proof-nets $\mathcal{R}$ of $\vdash A^\perp, B$ and $\mathcal{S}$ of $\vdash B^\perp, A$ such that
$\mathcal{R}$ cut with $\mathcal{S}$ reduces to id

General case difficult:

$$R_4 = X_1^+ \otimes X_2^+ \otimes X_3^+ \otimes X_4^+ \trianglelefteq (X_1^+ \otimes X_2^+ \otimes X_3^+ \otimes X_4^+) \parr (X_1^+ \otimes (X_1^- \parr (X_2^+ \otimes (X_2^- \parr (X_3^+ \otimes (X_3^- \parr (X_4^+ \otimes X_4^-)))))))$$

Retractions in proof-nets

In MLL Proof-Nets: retraction $A \trianglelefteq B$

Proof-nets $\mathcal{R}$ of $\vdash A^\perp, B$ and $\mathcal{S}$ of $\vdash B^\perp, A$ such that
$\mathcal{R}$ cut with $\mathcal{S}$ reduces to id

General case difficult:

$$R_4 = X_1^+ \otimes X_2^+ \otimes X_3^+ \otimes X_4^+ \trianglelefteq (X_1^+ \otimes X_2^+ \otimes X_3^+ \otimes X_4^+) \parr (X_1^+ \otimes (X_1^- \parr (X_2^+ \otimes (X_2^- \parr (X_3^+ \otimes (X_3^- \parr (X_4^+ \otimes X_4^-)))))))$$

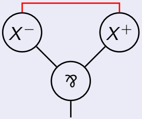
In MLL Proof-Nets: atomic retraction $X^+ \trianglelefteq B$

Proof-nets $\mathcal{R}$ of $\vdash X^-, B$ and $\mathcal{S}$ of $\vdash B^\perp, X^+$ ~~such that~~
~~$\mathcal{R}$ cut with $\mathcal{S}$ reduces to id~~

Key Result: finding a shape

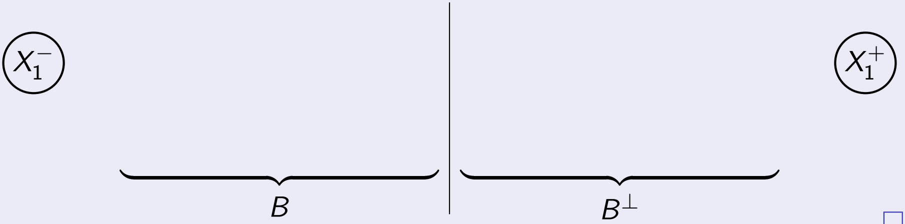
Lemma

In $X^+ \trianglelefteq B$ one of the two proof-nets contains



Proof.

Follow a GOI path until finding it

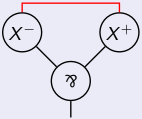


□

Key Result: finding a shape

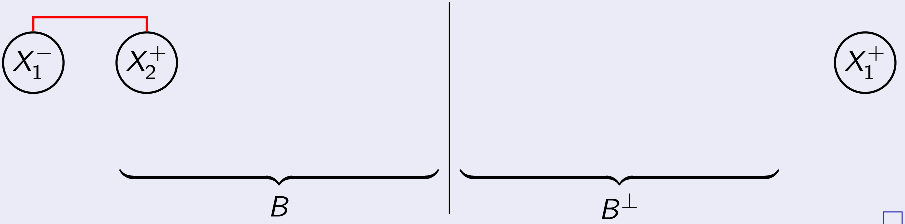
Lemma

In $X^+ \trianglelefteq B$ one of the two proof-nets contains



Proof.

Follow a GOI path until finding it

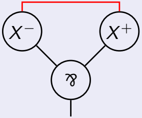


□

Key Result: finding a shape

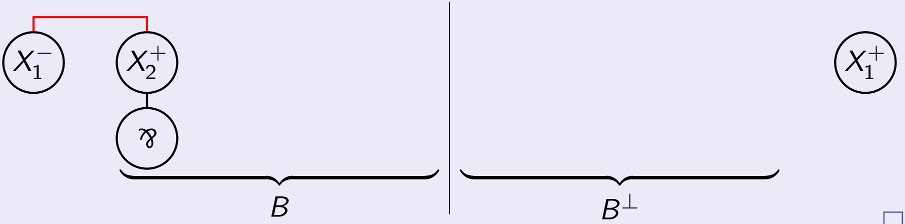
Lemma

In $X^+ \trianglelefteq B$ one of the two proof-nets contains



Proof.

Follow a GOI path until finding it

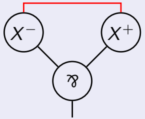


□

Key Result: finding a shape

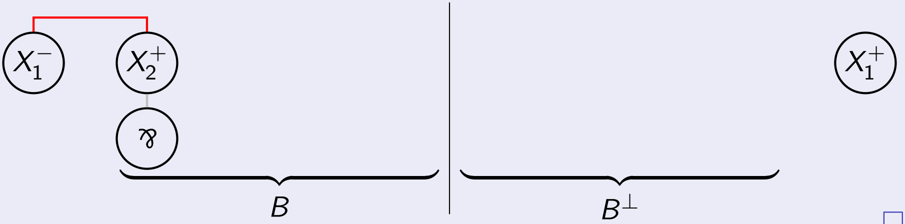
Lemma

In $X^+ \trianglelefteq B$ one of the two proof-nets contains



Proof.

Follow a GOI path until finding it

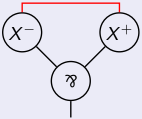


□

Key Result: finding a shape

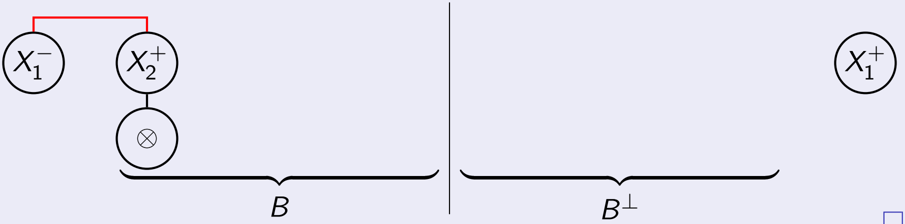
Lemma

In $X^+ \trianglelefteq B$ one of the two proof-nets contains



Proof.

Follow a GOI path until finding it

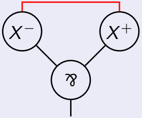


□

Key Result: finding a shape

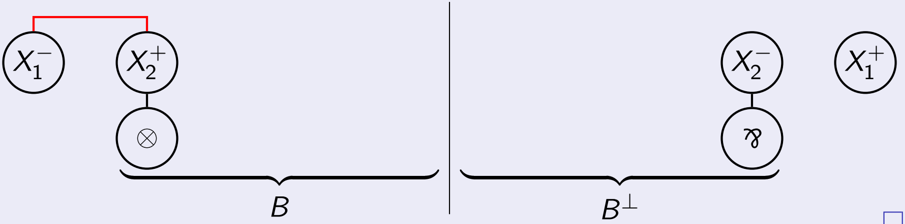
Lemma

In $X^+ \trianglelefteq B$ one of the two proof-nets contains



Proof.

Follow a GOI path until finding it

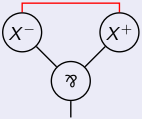


□

Key Result: finding a shape

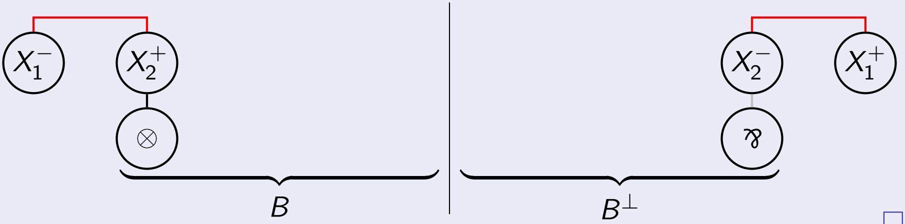
Lemma

In $X^+ \trianglelefteq B$ one of the two proof-nets contains



Proof.

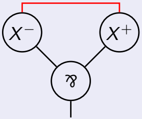
Follow a GOI path until finding it



Key Result: finding a shape

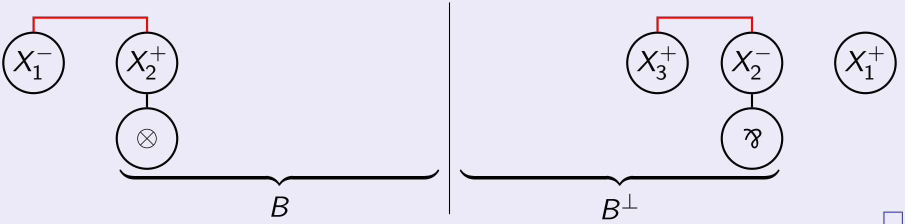
Lemma

In $X^+ \trianglelefteq B$ one of the two proof-nets contains



Proof.

Follow a GOI path until finding it

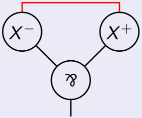


□

Key Result: finding a shape

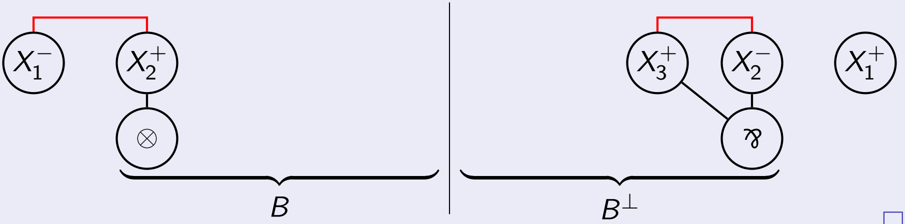
Lemma

In $X^+ \trianglelefteq B$ one of the two proof-nets contains



Proof.

Follow a GOI path until finding it

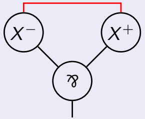


□

Key Result: finding a shape

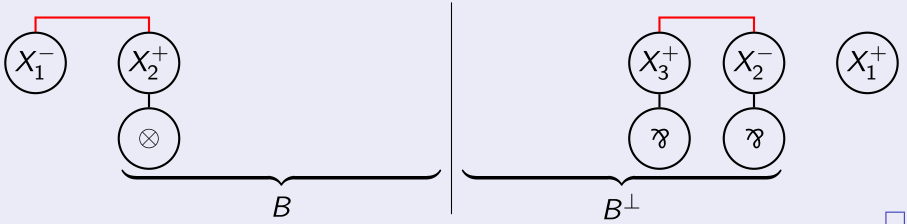
Lemma

In $X^+ \trianglelefteq B$ one of the two proof-nets contains



Proof.

Follow a GOI path until finding it

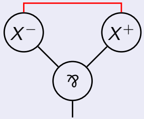


□

Key Result: finding a shape

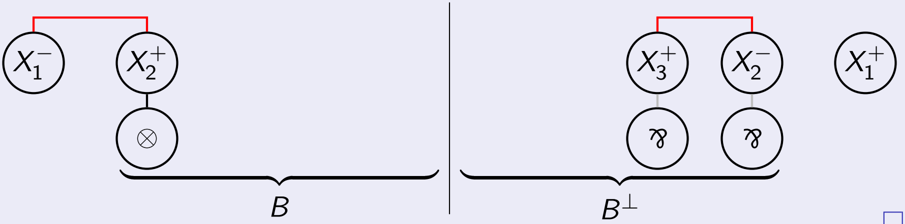
Lemma

In $X^+ \trianglelefteq B$ one of the two proof-nets contains



Proof.

Follow a GOI path until finding it

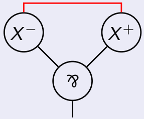


□

Key Result: finding a shape

Lemma

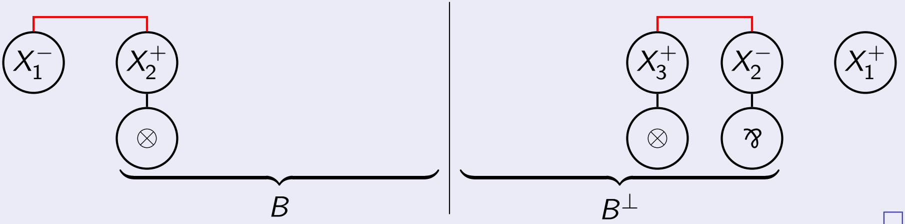
In $X^+ \trianglelefteq B$ one of the two proof-nets contains



Proof.

Follow a GOI path until finding it

<u>Invariant:</u> every X^+ of B is above a $\otimes$, and every X^- above a ⅋

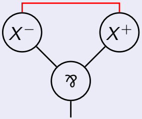


□

Key Result: finding a shape

Lemma

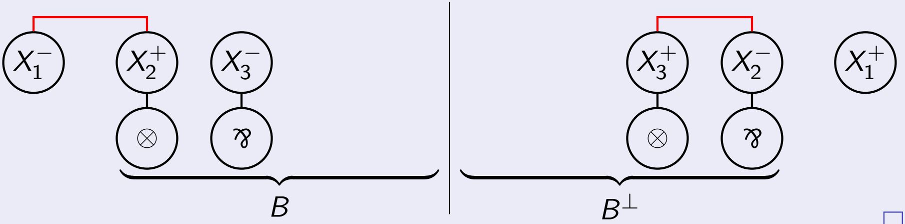
In $X^+ \trianglelefteq B$ one of the two proof-nets contains



Proof.

Follow a GOI path until finding it

Invariant: every X^+ of B is above a $\otimes$, and every X^- above a $\parr$

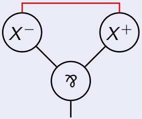


□

Key Result: finding a shape

Lemma

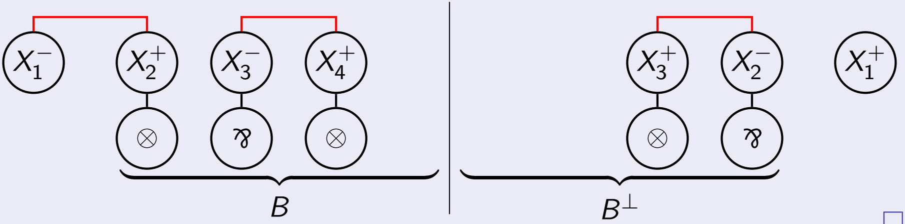
In $X^+ \trianglelefteq B$ *one of the two proof-nets contains*



Proof.

Follow a GOI path until finding it

<u>Invariant:</u> every X^+ of B is above a $\otimes$, and every X^- above a ⅋

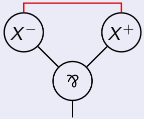


□

Key Result: finding a shape

Lemma

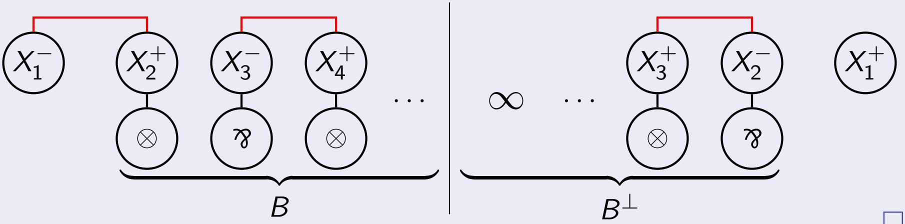
In $X^+ \trianglelefteq B$ *one of the two proof-nets contains*



Proof.

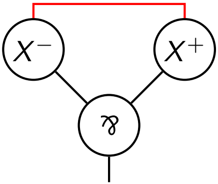
Follow a GOI path until finding it

Invariant: every X^+ of B is above a $\otimes$, and every X^- above a ⅋

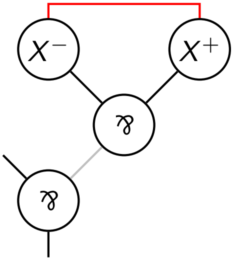


□

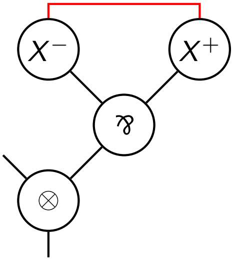
Using the found pattern



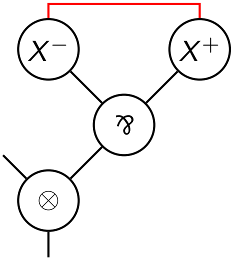
Using the found pattern

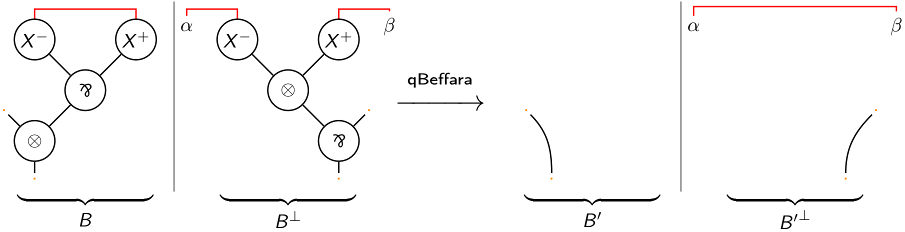


Using the found pattern

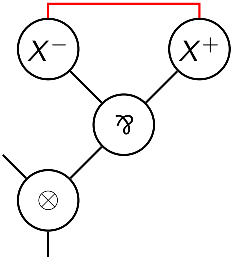


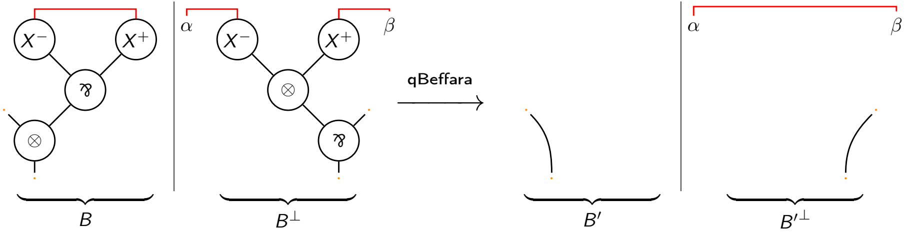
Using the found pattern





Using the found pattern





$\longrightarrow$ This rewriting **preserves** being a retraction!

Atomic retractions

Theorem

$X^+ \trianglelefteq B$ iff B is obtained from X^+ by Beffara $A \trianglelefteq A \otimes (A^\perp \parr A)$ and isomorphisms

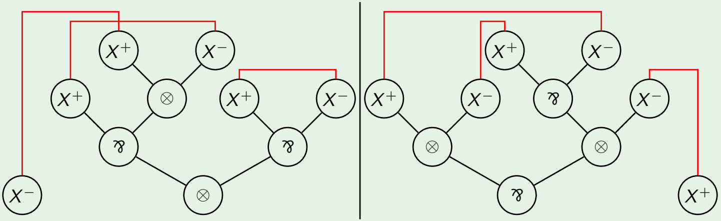
Atomic retractions

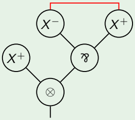
Theorem

$X^+ \trianglelefteq B$ iff B is obtained from X^+ by Beffara $A \trianglelefteq A \otimes (A^\perp \parr A)$ and isomorphisms

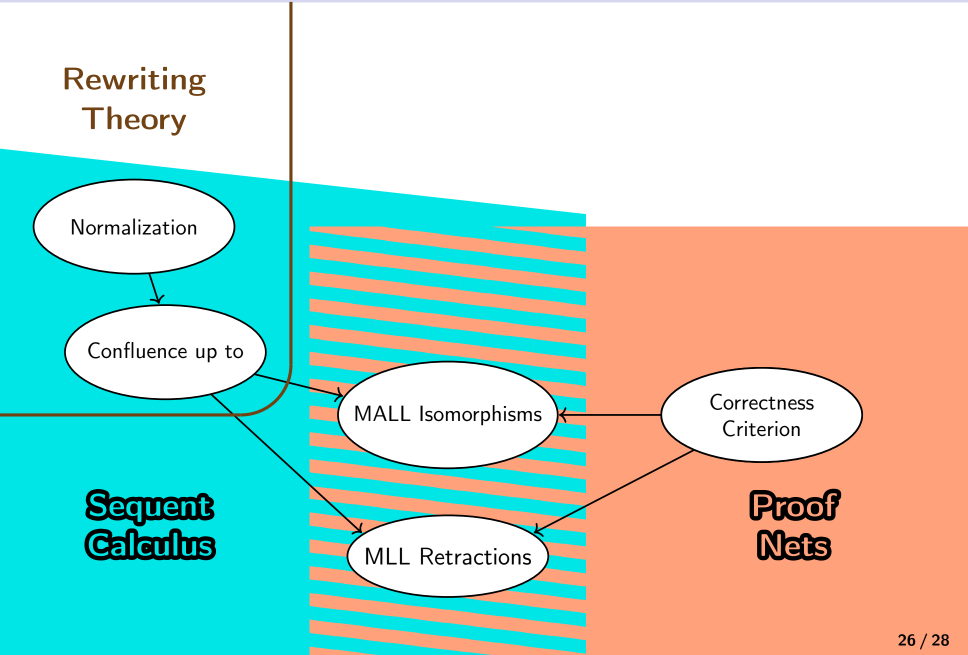
…but only at the level of **formulas**; Beffara does not give all **proofs**!

Proofs of $X^+ \trianglelefteq (X^+ \otimes X^-) \parr ((X^+ \parr X^-) \otimes X^-)$

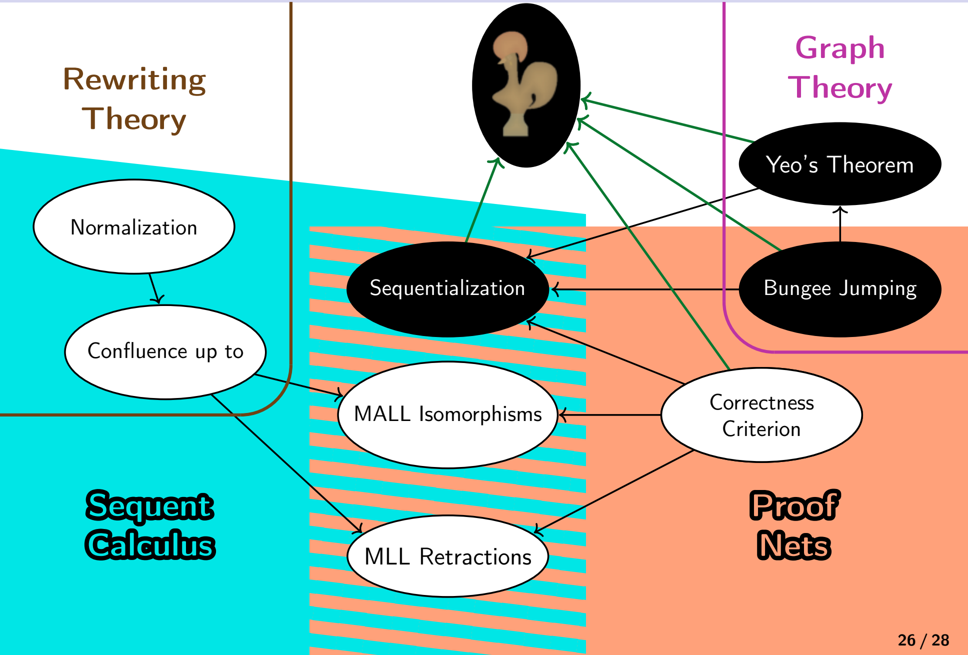


no  so cannot be generated!

Thesis' Overview

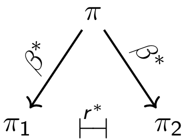


Thesis' Overview



Future Work

- We should have all tools to expend the confluence up to result to LL



- Isomorphisms for MELL or MALL with 1st order quantifiers (proof-nets)
- Characterize all retractions in MLL

Merci !

References I

[BD99] Vincent Balat and Roberto Di Cosmo. "A Linear Logical View of Linear Type Isomorphisms". In: *Computer Science Logic*. Ed. by Jörg Flum and Mario Rodríguez-Artalejo. Vol. 1683. Lecture Notes in Computer Science. Springer, 1999, pp. 250–265.

[CP05] Robin Cockett and Craig Pastro. "A Language For Multiplicative-additive Linear Logic". In: *Electronic Notes in Theoretical Computer Science* 122 (2005). Proceedings of the 10th Conference on Category Theory in Computer Science (CTCS 2004), pp. 23–65. DOI: /10.1016/j.entcs.2004.06.049. URL: https://www.sciencedirect.com/science/article/pii/S1571066105000320.

References II

[GH83] Jerrold W. Grossman and Roland Häggkvist. “Alternating Cycles in Edge-Partitioned Graphs”. In: *Journal of Combinatorial Theory, Series B* 34.1 (1983), pp. 77–81. ISSN: 0095-8956. DOI: 10.1016/0095-8956(83)90008-4. URL: https://www.sciencedirect.com/science/article/pii/0095895683900084.

[Gir87] Jean-Yves Girard. “Linear logic”. In: *Theoretical Computer Science* 50 (1987), pp. 1–102. DOI: 10.1016/0304-3975(87)90045-4.

[HG05] Dominic Hughes and Rob van Glabbeek. “Proof Nets for Unit-free Multiplicative-Additive Linear Logic”. In: *ACM Transactions on Computational Logic* 6.4 (2005), pp. 784–842. DOI: 10.1145/1094622.1094629.

References III

[Hue80] Gérard Huet. "Confluent Reductions: Abstract Properties and Applications to Term Rewriting Systems: Abstract Properties and Applications to Term Rewriting Systems". In: *Journal of the ACM* 27.4 (Oct. 1980), pp. 797–821. DOI: 10.1145/322217.322230.

[Kot59] Anton Kotzig. "On the theory of finite graphs with a linear factor. II.". slo. In: *Matematicko-Fyzikálny Časopis* 09.3 (1959). In Slovak, with as original title Z teórie konečných grafov s lineárnym faktorom. II., pp. 136–159. URL: https://eudml.org/doc/29908.

[Lau05] Olivier Laurent. "Classical isomorphisms of types". In: *Mathematical Structures in Computer Science* 15.5 (Oct. 2005), pp. 969–1004.

References IV

[Ngu20] Lê Thành Dũng Nguyễn. "Unique perfect matchings, forbidden transitions and proof nets for linear logic with Mix". In: *Logical Methods in Computer Science* 16.1 (Feb. 2020). DOI: 10.23638/LMCS-16(1:27)2020.

[Sey78] Paul D. Seymour. "Sums of circuits". In: *Graph Theory and Related Topics* (1978). Ed. by J. A. Bondy and U. S. R. Murty, pp. 341–355.

[Sol83] Sergei Soloviev. "The category of finite sets and cartesian closed categories". In: *Journal of Soviet Mathematics* 22.3 (1983), pp. 1387–1400.

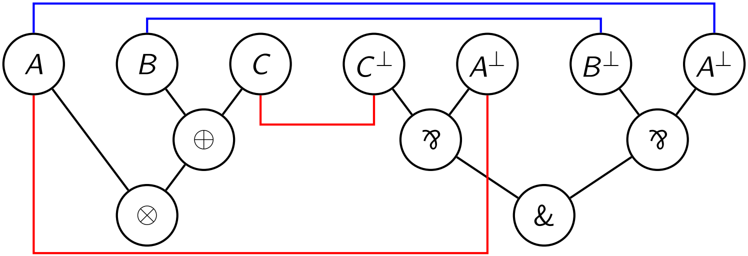
References V

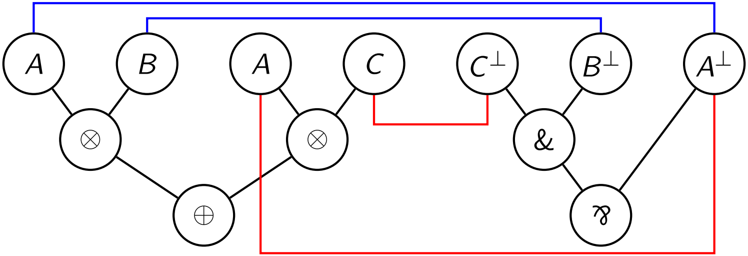
[SS79] D. J. Shoesmith and T. J. Smiley. "Theorem on Directed Graphs, Applicable to Logic". In: *Journal of Graph Theory* 3.4 (1979), pp. 401–406. DOI: 10.1002/jgt.3190030412. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/jgt.3190030412.

[Sze04] Stefan Szeider. "On Theorems Equivalent with Kotzig's Result on Graphs with Unique 1-Factors". In: *Ars Combinatoria* 73 (2004), pp. 53–64. URL: https://www.ac.tuwien.ac.at/files/pub/szeider-AC-2004.pdf.

[Yeo97] Anders Yeo. "A Note on Alternating Cycles in Edge-Coloured Graphs". In: *Journal of Combinatorial Theory, Series B* 69.2 (1997), pp. 222–225. DOI: 10.1006/jctb.1997.1728.

Proof-nets for $A \otimes (B \oplus C) \simeq (A \otimes B) \oplus (A \otimes C)$

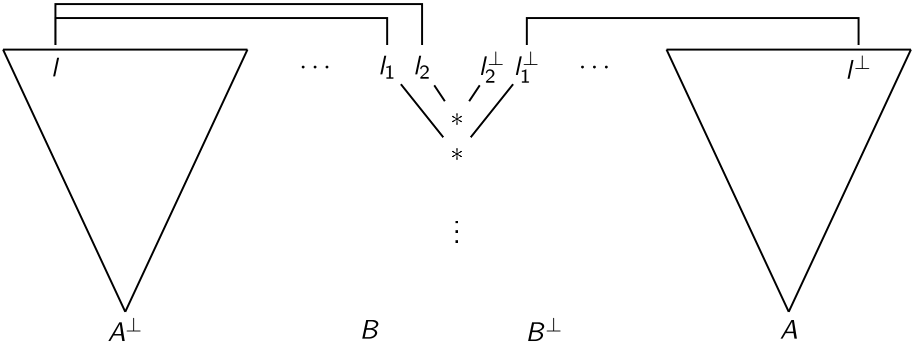




Proof 3/3: Why the distributed shape?

$A \otimes (B \oplus C) \simeq (A \otimes B) \oplus (A \otimes C)$ **not** of this shape

Correctness criterion to get this local shape from global *distributivity*

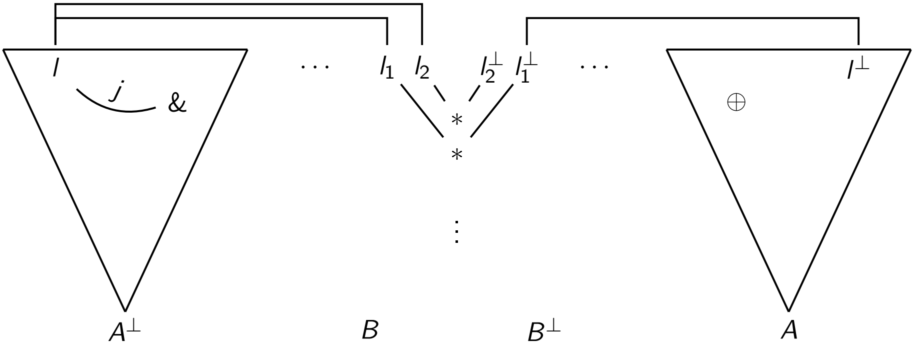


1. Forbidden configuration

Proof 3/3: Why the distributed shape?

$A \otimes (B \oplus C) \simeq (A \otimes B) \oplus (A \otimes C)$ **not** of this shape

Correctness criterion to get this <u>local</u> shape from <u>global</u> *distributivity*

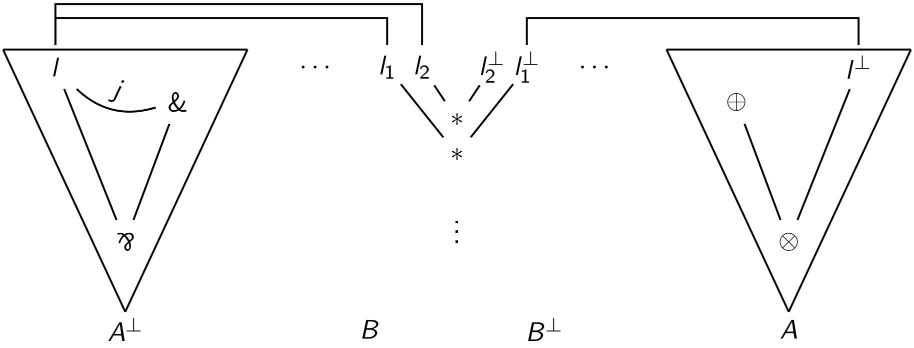


1. Forbidden configuration
2. Dependence on a &

Proof 3/3: Why the distributed shape?

$A \otimes (B \oplus C) \simeq (A \otimes B) \oplus (A \otimes C)$ **not** of this shape

Correctness criterion to get this local shape from global *distributivity*

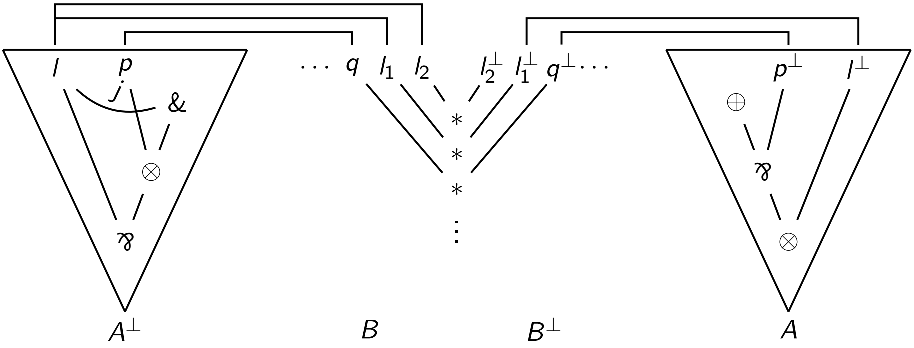


1. Forbidden configuration
2. Dependence on a &
3. ⅋ below

Proof 3/3: Why the distributed shape?

$A \otimes (B \oplus C) \simeq (A \otimes B) \oplus (A \otimes C)$ **not** of this shape

Correctness criterion to get this local shape from global *distributivity*



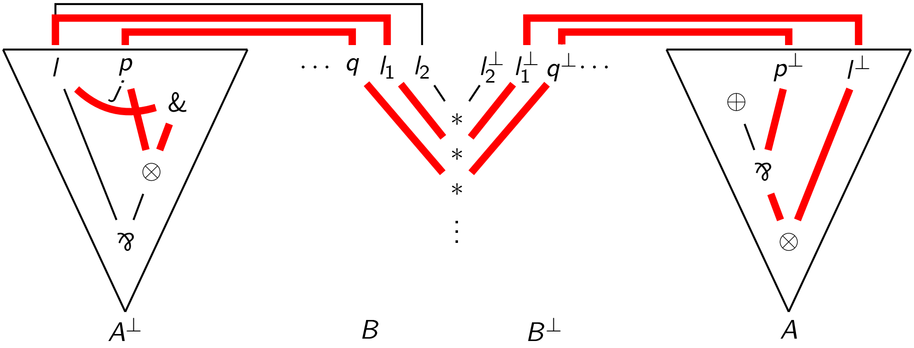
1. Forbidden configuration
2. Dependence on a &
3. ⅋ below

4. Distributivity

Proof 3/3: Why the distributed shape?

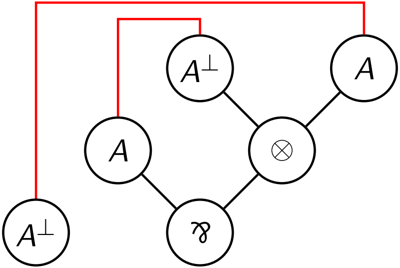
$A \otimes (B \oplus C) \simeq (A \otimes B) \oplus (A \otimes C)$ **not** of this shape

Correctness criterion to get this local shape from global *distributivity*

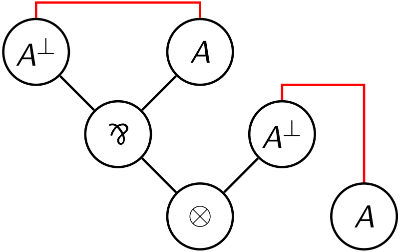


1. Forbidden configuration
2. Dependence on a &
3. ⅋ below
4. Distributivity
5. Forbidden cycle

Beffara $A \trianglelefteq A \otimes (A^\perp \parr A)$ is a retraction



$A \parr (A^\perp \otimes A)$



$(A^\perp \parr A) \otimes A^\perp$

Around Sequentialization

Sequentialization [HG05]

__MALL__ Proof-nets are exactly the images of proofs.

Around Sequentialization

Sequentialization [Gir87]

MLL *Proof-nets are exactly the images of proofs.*

↑

Sequentialization [HG05]

MALL *Proof-nets are exactly the images of proofs.*

Around Sequentialization

Sequentialization [Gir87]

MLL *Proof-nets are exactly the images of proofs.*

[Ngu20]
encoding

Kotzig [Kot59]

On perfect matchings

Sequentialization [HG05]

MALL *Proof-nets are exactly the images of proofs.*

Proof-nets

Graph Theory

Around Sequentialization

Sequentialization [Gir87]

***MLL** Proof-nets are exactly the images of proofs.*

↑

Sequentialization [HG05]

***MALL** Proof-nets are exactly the images of proofs.*

Proof-nets

[Ngu20]
↔ encoding

all equivalent using encodings [Sze04]

Kotzig [Kot59]

On perfect matchings

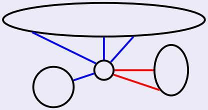
Grossman & Häggkvist [GH83]

Seymour & Giles [Sey78]

Shoesmith & Smiley [SS79]

Yeo [Yeo97]

A graph with no alternating cycle has a splitting vertex:



Graph Theory

Around Sequentialization

Proof-nets

Sequentialization [Gir87]

MLL *Proof-nets are exactly the images of proofs.*

Sequentialization [HG05]

MALL *Proof-nets are exactly the images of proofs.*

Graph Theory

all equivalent using encodings [Sze04]

Kotzig [Kot59]

On perfect matchings

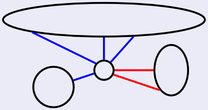
Grossman & Häggkvist [GH83]

Seymour & Giles [Sey78]

Shoesmith & Smiley [SS79]

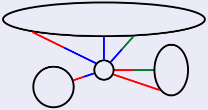
Yeo [Yeo97]

A graph with no alternating cycle has a splitting vertex:



encoding

Yeo with local coloring



(and a parameter)

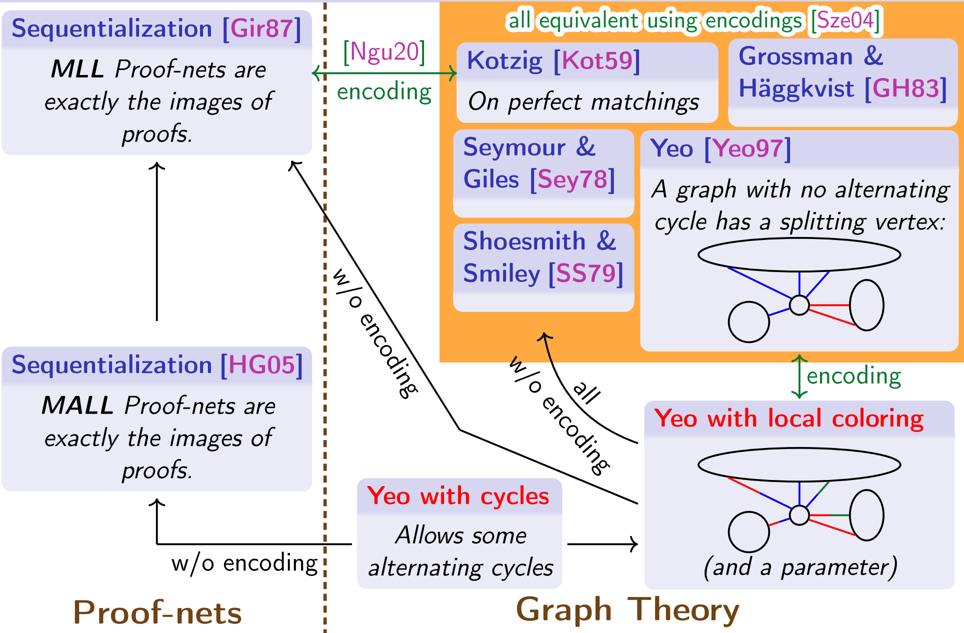
[Ngu20] encoding

w/o encoding

all

w/o encoding

Around Sequentialization



Bungee Jumping