

An introduction to Linear Logic

Jean-Baptiste Joinet

IRPhIL (Univ. Jean Moulin Lyon 3) et IHPST

07/06/2022

Linear Logic: 36 years from birth (1986) to our days (2022)

2022: the words 'Linear Logic' cover a huge field

- ▶ Various chapters and tools, among which:
 - ▶ Sequent calculi
 - ▶ Proofs-nets
 - ▶ Denotational semantics
 - ▶ Game semantics and Ludics
 - ▶ Geometry of interaction
 - ▶ Implicit computational complexity
 - ▶ Categorical approaches
 - ▶ etc
- ▶ Various proofs systems
 - ▶ fragments, variants, extensions of the original LL : e.g. MLL, ALL, MALL, MELL, LL, ELL, LLL etc
 - ▶ Linear Logics

Linear Logic: 36 years from birth (1986) to our days (2022)

1986: birth of LL

- ▶ Due to Jean-Yves Girard
- ▶ LL borns at the meeting point of
 - ▶ two scientific lines:
 - ▶ *Computing theory* (Denotational semantics of lambda-calculus)
 - ▶ *Proof theory* (Logic)
 - ▶ linked by a bridge: Curry-Howard isomorphism (proofs analytization = programs execution)

What denoted the words “Linear Logic” at the very beginning?

- ▶ A sequent calculus logical system
- ▶ “Yet another formal logic” ? No.
- ▶ **Linear Logic = Classical logic, but decomposed and observed through the microscope of the computational point of view on proofs**
- ▶ Goal of this introduction: make understand what this means. . .

Linear Logic: 36 years from birth (1986) to our days (2022)

PLAN :

- ▶ PART 1. Lambda-Calculus: a new horizon for semantics
- ▶ PART 2. Proofs in classical logic (static)
- ▶ PART 3. The Linear decomposition of Classical/Intuitionistic Logic
- ▶ PART 4. Proof nets
- ▶ PART 5. Ali Baba's cave : the variety of exponentials
- ▶ PART 6. From Proof nets to the Geometry of interaction to Transcendental syntax
- ▶ PART 7. From Linear sequent calculi to Ludics

PART 1

What is Lambda calculus ?

Trying to answering from a “Logic and Interdisciplinarity” viewpoint

- ▶ Explained to a mathematician ?
- ▶ Explained to a computer scientist ?
- ▶ Explained to a philosopher?

What is Lambda calculus ?

Trying to answering from a “Logic and Interdisciplinarity” viewpoint

- ▶ Explained to a mathematician ?

Historically, an attempt to refound Logic from the notion of function (after the antinomies in Cantorian/Fregean logic)

- ▶ Explained to a computer scientist ?

- ▶ Explained to a philosopher?

What is Lambda calculus ?

Trying to answering from a “Logic and Interdisciplinarity” viewpoint

- ▶ Explained to a mathematician ?

Historically, an attempt to refound Logic from the notion of function (after the antinomies in Cantorian/Fregean logic)

- ▶ Explained to a computer scientist ?

A programming language

- ▶ Explained to a philosopher?

What is Lambda calculus ?

Trying to answering from a “Logic and Interdisciplinarity” viewpoint

- ▶ Explained to a mathematician ?

Historically, an attempt to refound Logic from the notion of function (after the antinomies in Cantorian/Fregean logic)

- ▶ Explained to a computer scientist ?

A programming language

- ▶ Explained to a philosopher?

A framework to reconstruct the logical-mathematical semantics/ontology from the idea that individuals are first defined by their actional (or interactional) behavior in time.

Lambda-calculus: general idea

1936: Alonzo Church defines the λ -calculus

2022 : what is called today (pure) λ -calculus = a kernel fragment of Church's original system

Lambda-calculus: general idea

1936: Alonzo Church defines the λ -calculus

2022 : what is called today (pure) λ -calculus = a kernel fragment of Church's original system

General idea (pedagogical first approximation): it's a naive theory of functions, where **everything is functional** (later on, we will more adequately characterize it as a frame able to represent algorithms)

Concretely :

- ▶ it starts with a language to design functional individuals terms (the terms of λ -calculus a.k.a. λ -terms)
- ▶ but the way those terms are built **extend** in several respects the ways first order functional "individuals terms" are built

Lambda-calculus: general idea

1936: Alonzo Church defines the λ -calculus

2022 : what is called today (pure) λ -calculus = a kernel fragment of Church's original system

General idea (pedagogical first approximation): it's a naive theory of functions, where **everything is functional** (later on, we will more adequately characterize it as a frame able to represent algorithms)

Concretely :

- ▶ it starts with a language to design functional individuals terms (the terms of λ -calculus a.k.a. λ -terms)
- ▶ but the way those terms are built **extend** in several respects the ways first order functional "individuals terms" are built

Let us recall that first order individual terms are built by applying function constants having a given arity to the relevant number of arguments – being themselves terms: e.g. $f(g(x), g(c))$

Let us see in which respects λ -terms differ from first order

Our first way to build λ -terms: λ -abstraction

One major innovation/extension is that the new terms (λ -terms) include “definite descriptions” (cf. B. Russell, *On denoting*, 1905), a construction absent from first order terms construction.

In the everyday mathematical language, definite descriptions are a common way to designate functions, e.g:

Our first way to build λ -terms: λ -abstraction

One major innovation/extension is that the new terms (λ -terms) include “definite descriptions” (cf. B. Russell, *On denoting*, 1905), a construction absent from first order terms construction.

In the everyday mathematical language, definite descriptions are a common way to designate functions, e.g:

- ▶ “The function $2x$ is monotonic”
- ▶ i.e. “The function which, to x , associates $2x$ is monotonic”
(is a definite description)

Our first way to build λ -terms: λ -abstraction

One major innovation/extension is that the new terms (λ -terms) include “definite descriptions” (cf. B. Russell, *On denoting*, 1905), a construction absent from first order terms construction.

In the everyday mathematical language, definite descriptions are a common way to designate functions, e.g:

- ▶ “The function $2x$ is monotonic”
- ▶ i.e. “The function which, to x , associates $2x$ is monotonic”
(is a definite description)
- ▶ “Let $f : x \mapsto 2x$. The function f is monotonic”.
- ▶ “ $x \mapsto 2x$ is monotonic”.

Our first way to build λ -terms: λ -abstraction

One major innovation/extension is that the new terms (λ -terms) include “definite descriptions” (cf. B. Russell, *On denoting*, 1905), a construction absent from first order terms construction.

In the everyday mathematical language, definite descriptions are a common way to designate functions, e.g:

- ▶ “The function $2x$ is monotonic”
- ▶ i.e. “The function which, to x , associates $2x$ is monotonic”
(is a definite description)
- ▶ “Let $f : x \mapsto 2x$. The function f is monotonic”.
- ▶ “ $x \mapsto 2x$ is monotonic”.

Using λ -calculus notation, $x \mapsto 2x$ would be noted $\lambda x 2x$.

More generally, if t is a λ -term, then $\lambda x t$ also will be a λ -term
(provisionally read as: **the function which to x associates t**).

Our second way to build λ -terms: application

The second way to build λ -terms, which follows the idea of **applying a function to an argument**, is by **applying a term to a term**.

This is **almost similar** to first order terms construction when, for instance, we apply the unary function constant g to the term u , forming the first order term $g(u)$.

Our second way to build λ -terms: application

The second way to build λ -terms, which follows the idea of **applying a function to an argument**, is by **applying a term to a term**.

This is **almost similar** to first order terms construction when, for instance, we apply the unary function constant g to the term u , forming the first order term $g(u)$.

The **notation** of “application” however diverges:

Notation: when applying the λ -term t to an argument u (a λ -term too):

- ▶ we will not write $t(u)$
- ▶ we will write $(t\ u)$

Our second way to build λ -terms: application

The second way to build λ -terms, which follows the idea of **applying a function to an argument**, is by **applying a term to a term**.

This is **almost similar** to first order terms construction when, for instance, we apply the unary function constant g to the term u , forming the first order term $g(u)$.

The **notation** of “application” however diverges:

Notation: when applying the λ -term t to an argument u (a λ -term too):

▶ we will not write $t(u)$

▶ we will write $(t\ u)$

Why did we say **almost similar** (beyond the detail of the notational divergence)?

Pursuing the comparison λ -terms/terms of first order logic

In first order logic, function constants born with an arity (i.e. a given 'legal' number of arguments): e.g. to build the term $f(g(x), g(c))$ is correct only if f 's arity is 2 (binary function constant), g 's arity is 1 (unary function) and c 's arity is 0 (individual constant)

Pursuing the comparison λ -terms/terms of first order logic

In first order logic, function constants born with an arity (i.e. a given 'legal' number of arguments): e.g. to build the term $f(g(x), g(c))$ is correct only if f 's arity is 2 (binary function constant), g 's arity is 1 (unary function) and c 's arity is 0 (individual constant)

In λ -calculus, a (function) variable has no arity (better said: **it is simultaneously of any arity**). The fact that, in Λ , the application term constructor $(t u)$, formally allows to "give" only one argument to a function variable f – as when one forms $(f u)$ –, must not be interpreted as if "only 1-ary functions" were involved.

Pursuing the comparison λ -terms/terms of first order logic

In first order logic, function constants born with an arity (i.e. a given 'legal' number of arguments): e.g. to build the term $f(g(x), g(c))$ is correct only if f 's arity is 2 (binary function constant), g 's arity is 1 (unary function) and c 's arity is 0 (individual constant)

In λ -calculus, a (function) variable has no arity (better said: **it is simultaneously of any arity**). The fact that, in Λ , the application term constructor $(t u)$, formally allows to "give" only one argument to a function variable f – as when one forms $(f u)$ –, must not be interpreted as if "only 1-ary functions" were involved. **Applying f to k arguments may indeed be "encoded" by k successive unary applications:**

- ▶ f f is 0-ary
- ▶ $(f u_1)$ f is also 1-ary
- ▶ $((f u_1) u_2)$ f is also 2-ary
- ▶ \vdots
- ▶ $(\dots((f u_1) u_2)\dots u_k)$ f is also k -ary

Convention: $(\dots((t u_1) u_2)\dots u_k)$ will be noted $t u_1 u_2 \dots u_k$

Definition of lambda-terms

To sum up: inductive definition of λ -terms:

1. **variables:**

$x, y, \dots, f, g \dots$ are λ -terms

2. **λ -abstraction:**

$\lambda x t$ is a λ -term, if t is a λ -term and x is a variable

3. **application)**

$(t u)$ is a λ -term, if t and u are λ -terms.

Examples:

$f \quad x \quad (f x) \quad ((f x)y) \quad (f(x y)) \quad \lambda x x$
 $(\lambda x x \lambda y y) \quad \lambda x (x x) \quad \lambda x (\lambda f \lambda y (f (\lambda y y (x f)))) \lambda z z$

Which references for λ -terms ?

Some λ -terms can easily be interpreted. For example:

$\lambda x x$ = the function which to any x associates x
= **the identity function**

Which references for λ -terms ?

Some λ -terms can easily be interpreted. For example:

$\lambda x x$ = the function which to any x associates x
= **the identity function**

Some other λ -terms could have a vague reference (set-theoretically, say), but a reference which remains undetermined. Examples :

- ▶ f denotes a function, but which one ?
- ▶ $(f x)$ denotes a function applied to an argument, but which one?

Those terms, f or $(f x)$, have an undetermined reference because, contrary to the previous example ($\lambda x x$), **they contain free variables** (they are non closed terms).

Which references for λ -terms ?

Some λ -terms can easily be interpreted. For example:

$\lambda x x$ = the function which to any x associates x
= **the identity function**

Some other λ -terms could have a vague reference (set-theoretically, say), but a reference which remains undetermined. Examples :

- ▶ f denotes a function, but which one ?
- ▶ $(f x)$ denotes a function applied to an argument, but which one?

Those terms, f or $(f x)$, have an undetermined reference because, contrary to the previous example ($\lambda x x$), **they contain free variables** (they are non closed terms).

λx is a binder (for variable occurrences in λ -terms), exactly as quantifiers (e.g. $\forall x$) are binders (for variable occurrences in formulas). Compare $\lambda x x$ and $\forall x P x$.

Focus on free variables, closed λ -terms, alpha-equivalence

Definition

- ▶ An occurrence of x in t , which is in the scope of a λx , is a **bound occurrence** of x in t
- ▶ A variable is **free** in t , if it has an occurrence in t which is not bound of t

Example: $t = \lambda x((x y)\lambda y((y x) y))$ (y is free in t ; but x not)

Definition

t is a **closed term**, if it has no free variable

Focus on free variables, closed λ -terms, alpha-equivalence

Definition

- ▶ An occurrence of x in t , which is in the scope of a λx , is a **bound occurrence** of x in t
- ▶ A variable is **free** in t , if it has an occurrence in t which is not bound of t

Example: $t = \lambda x((x y)\lambda y((y x) y))$ (y is free in t ; but x not)

Definition

t is a **closed term**, if it has no free variable

As usual with binding, 'names' of bound variables doesn't matter, e.g. any of $\lambda x x$ and $\lambda y y$ denotes the identity function

Definition

Terms which (approximately said) differ only w.r.t. the "name" of their bound variables are said **α -equivalent** (notation: \approx_α).

Example: $\lambda x \lambda y (y x) \approx_\alpha \lambda z \lambda y (y z) \approx_\alpha \lambda y \lambda x (x y)$

From now on, $\Lambda =$ the set of λ -terms up to α -equivalence

Irrelevance of the usual set-theoretical interpretation

We saw that (because they have free variables), λ -terms as f or $(f x)$ could only be interpreted in an underdetermined way.

But some terms cannot have even a vague (set-theoretical) reference. For instance, which reference for $(f f)$?

From a set-theoretical viewpoint, applying a function to itself has no meaning, not even an 'underdetermined' one.

Irrelevance of the usual set-theoretical interpretation

We saw that (because they have free variables), λ -terms as f or $(f\ x)$ could only be interpreted in an underdetermined way.

But some terms cannot have even a vague (set-theoretical) reference. For instance, which reference for $(f\ f)$?

From a set-theoretical viewpoint, applying a function to itself has no meaning, not even an 'underdetermined' one.

Moreover, the usual set-theoretical reading available for first-order terms, is just impossible for such a λ -term:

- ▶ in $(f\ f)$, the variable f is both function and argument
- ▶ the full set-theoretical interpretation would require, for some set X , to identify (to render isomorphic) the sets X and X^X (the set of all functions from X to X)
- ▶ this is however impossible for cardinality reasons (at least if one keeps **all** the functions from X to X) since $\#(X) \neq \#(X^X)$.

Two ways to face this lost of interpretability

WAY 1: GETTING RID OF UNINTERPRETABLE TERMS (1/2)

If terms like $(f f)$ are non interpretable, let's get rid of them!

A Russellian idea: sometimes, applying a function to an argument should be prohibited. Functions and arguments are not free (pure) objects: they are of a given type (which defines their "essence"): the application of a term to a term should thus be allowed only if their essence (their types) are adequate for that composition.

A set-theoretical approximation of that idea: a function f has a domain, say the set A , and a codomain, say the set B (usual notation $f : A \rightarrow B$) and one should write $f(x)$ only if x belongs to A .

Two ways to face this lost of interpretability

WAY 1: GETTING RID OF UNINTERPRETABLE TERMS (1/2)

If terms like $(f f)$ are non interpretable, let's get rid of them!

A Russellian idea: sometimes, applying a function to an argument should be prohibited. Functions and arguments are not free (pure) objects: they are of a given type (which defines their “essence”): the application of a term to a term should thus be allowed only if their essence (their types) are adequate for that composition.

A set-theoretical approximation of that idea: a function f has a domain, say the set A , and a codomain, say the set B (usual notation $f : A \rightarrow B$) and one should write $f(x)$ only if x belongs to A .

Church's proposal : “**simply typed λ -calculus**”. Let us give a quick presentation. . .

Two ways to face this lost of interpretability (continued)

WAY 1: GETTING RID OF UNINTERPRETABLE TERMS (2/2)

Concretely, Church introduces a **second level grammar** (the grammar of types):

- ▶ $X, Y \dots$ are types (atomic ones);
- ▶ $A \rightarrow B$ is a type, if A, B are types.

Example: $(X \rightarrow Z) \rightarrow ((Y \rightarrow X) \rightarrow Z)$

Two ways to face this lost of interpretability (continued)

WAY 1: GETTING RID OF UNINTERPRETABLE TERMS (2/2)

Concretely, Church introduces a **second level grammar** (the grammar of types):

- ▶ X, Y, \dots are types (atomic ones);
- ▶ $A \rightarrow B$ is a type, if A, B are types.

Example: $(X \rightarrow Z) \rightarrow ((Y \rightarrow X) \rightarrow Z)$

Once any λ -variable is given a (unique) type (superscript notation: x^A), a type is univocally inductively assigned to any term, in the following way:

$$\overbrace{\lambda x^A t^B}^{A \rightarrow B} \quad \text{and} \quad \overbrace{(t^{A \rightarrow B}) u^A}^B$$

Typically $(t)u$ will become possible only for $t^{A \rightarrow B}$ and u^A .

Two ways to face this lost of interpretability (continued)

WAY 1: GETTING RID OF UNINTERPRETABLE TERMS (2/2)

Concretely, Church introduces a **second level grammar** (the grammar of types):

- ▶ X, Y, \dots are types (atomic ones);
- ▶ $A \rightarrow B$ is a type, if A, B are types.

Example: $(X \rightarrow Z) \rightarrow ((Y \rightarrow X) \rightarrow Z)$

Once any λ -variable is given a (unique) type (superscript notation: x^A), a type is univocally inductively assigned to any term, in the following way:

$$\overbrace{\lambda x^A t^B}^{A \rightarrow B} \quad \text{and} \quad \overbrace{(t^{A \rightarrow B}) u^A}^B$$

Typically $(t)u$ will become possible only for $t^{A \rightarrow B}$ and u^A .

The typing process selects **a proper subset of Λ** (the set of λ -terms). In particular our queer term $(f f)$ will not be typable, hence will no more be considered as a correct term in that framework.

Two ways to face this lost of interpretability (continued)

WAY 2: GETTING RID OF SET-THEORETICAL SEMANTICS.

- ▶ second alternative : to consider a radically different kind of semantics, able to give a “meaning” even to queer λ -terms
- ▶ idea: **tell me how the term t acts, I will tell you who t is**
- ▶ Of course, one then needs to give a content to the idea that terms act, i.e. to **introduce action**. This will be done, via **β -reduction** of terms (a **dynamical** rewriting rule, to be defined soon)

Two ways to face this lost of interpretability (continued)

WAY 2: GETTING RID OF SET-THEORETICAL SEMANTICS.

- ▶ second alternative : to consider a radically different kind of semantics, able to give a “meaning” even to queer λ -terms
- ▶ idea: **tell me how the term t acts, I will tell you who t is**
- ▶ Of course, one then needs to give a content to the idea that terms act, i.e. to **introduce action**. This will be done, via **β -reduction** of terms (a **dynamical** rewriting rule, to be defined soon)
- ▶ But even at the **static** level, we can look at λ -terms through this actional reading – and characterize their intended action by a name:
 - ▶ $\lambda x x =$ the term which acts on any x , by giving it back = **the identity action**
 - ▶ $\lambda x \lambda f (f (f x)) =$ the term which takes any (argument) x , then any (action) f , and then makes f acting over x , and then makes f acting again on the result = **the iterator twice** (the doubly iterating operator)
 - ▶ $\lambda f (f f) =$ the term which takes any operation f and then make it operate on itself = **the auto-applicator**

Dynamic of λ -terms: β -reduction

Introducing action

A term $\lambda x u$ is ready to act when it is **confronted to an argument**, thus when it is in the context of an application: $(\lambda x u \ v)$

The way $\lambda x u$ acts is then defined via a rewriting rule : **the β -reduction**.

Dynamic of λ -terms: β -reduction

Introducing action

A term $\lambda x u$ is ready to act when it is **confronted to an argument**, thus when it is in the context of an application: $(\lambda x u v)$

The way $\lambda x u$ acts is then defined via a rewriting rule : **the β -reduction**.

Terminology, notation, definitions

- ▶ **β -redex in t** : any subterm of t , having form $(\lambda x u v)$
- ▶ $u[v/x]$: the term obtained by substituting every free occurrences of x in u by v .
- ▶ **β_1 -reduction step**: $\dots(\lambda x u v)\dots \rightsquigarrow_{\beta_1} \dots u[v/x]\dots$
- ▶ **β -reduction**: reflexive & transitive closure of $\rightsquigarrow_{\beta_1}$, i.e. **iteration k times of β_1 -reduction steps** ($k \in \mathbb{N}$).

Reduction strategies

In general, a term t contain more than one redexes. So, performing a β_1 -reduction step of t , means :

1. to choose in t a redex to be reduced (making this choice determine our “strategy” for t 's β_1 -reducing)
2. to perform then the substitution corresponding to that redex (the term one gets doing so, being called a β_1 -reduct of t : in general, as a term t contain more than one redexes, it has several distinct β_1 -reducts).

Reduction strategies

In general, a term t contain more than one redexes. So, performing a β_1 -reduction step of t , means :

1. to choose in t a redex to be reduced (making this choice determine our “strategy” for t 's β_1 -reducing)
2. to perform then the substitution corresponding to that redex (the term one gets doing so, being called a β_1 -reduct of t : in general, as a term t contain more than one redexes, it has several distinct β_1 -reducts).

The notion of strategy extends to β -reduction (the iteration of β_1 -reduction steps). **To apply a β -reduction strategy means to select, at each step, a redex in the term obtained at the previous step, when one iterates the β -reduction's process.**

To sum up: a given term t determines (not one reduction, but) a set of reductions, each “path” of reduction corresponding to what one calls a reduction strategy.

Normal/normalizable/strongly normalizable terms

Definitions

- ▶ **A normal term:** a term with no redex
- ▶ **A normal form of t :** a normal term u such that $t \rightsquigarrow_{\beta} u$ (u is the result of the computation performed starting from t)
- ▶ **A normalizable term:** a term with at least one normal form
- ▶ **A strongly normalizable term:** a term with no infinite $\rightsquigarrow_{\beta_1}$ reductions (i.e. every reduction leads to a normal form)

Normal/normalizable/strongly normalizable terms

Definitions

- ▶ **A normal term:** a term with no redex
- ▶ **A normal form of t :** a normal term u such that $t \rightsquigarrow_{\beta} u$ (u is the result of the computation performed starting from t)
- ▶ **A normalizable term:** a term with at least one normal form
- ▶ **A strongly normalizable term:** a term with no infinite $\rightsquigarrow_{\beta_1}$ reductions (i.e. every reduction leads to a normal form)

Remarks

- ▶ **Some terms are not normalizable.** Example: $(\Delta \Delta) =$
$$\frac{(\lambda x (x x) \quad \lambda x (x x))}{(\lambda x \quad t \quad u)} \rightsquigarrow_{\beta_1} (x x) [\lambda x (x x) / x] = (\lambda x (x x) \quad \lambda x (x x))$$

a redex its reduct
- ▶ **Some terms are strongly normalizable**, in particular all the simply typed ones (the true reason why Church actually considered them)
- ▶ **Some terms are normalizable but not strongly:**
 $t = (\lambda y x (\Delta \Delta))$. Depending on the redex chosen (the blue one or the red one), i.e. of the *strategy*: $t \rightsquigarrow_{\beta_1} x$ or $t \rightsquigarrow_{\beta_1} t$.

Confluence of β -reduction. Unicity of normal forms

Church-Rosser Property

If $t \rightsquigarrow_{\beta} t'$ and $t \rightsquigarrow_{\beta} t''$,
there exists u such that:
 $t' \rightsquigarrow_{\beta} u$ and $t'' \rightsquigarrow_{\beta} u$

Unicity of normal forms

When a term has a normal form, this normal form is unique (it does not depend on the chosen strategy)

Back to the semantical/ontological question

Frege's distinction : **Sense versus Denotation:**

$$2 + 2 \quad 4 \quad 2 \times 2$$

all have the same intended denotation (reference), even if they don't have the same sense.

Back to the semantical/ontological question

Frege's distinction : **Sense versus Denotation:**

$$2 + 2 \quad 4 \quad 2 \times 2$$

all have the same intended denotation (reference), even if they don't have the same sense.

Crucial remark: computation is:

- ▶ a transformation of sense
- ▶ which (hopefully) preserves denotation:

$$(2 + 2) \times (2 \times 2) \rightsquigarrow 4 \times (2 \times 2) \rightsquigarrow 4 \times 4 \rightsquigarrow 16$$

What become those ideas when applied to λ -calculus ?

Back to the semantical/ontological question

Frege's distinction : **Sense versus Denotation**:

$$2 + 2 \quad 4 \quad 2 \times 2$$

all have the same intended denotation (reference), even if they don't have the same sense.

Crucial remark: computation is:

- ▶ a transformation of sense
- ▶ which (hopefully) preserves denotation:

$$(2 + 2) \times (2 \times 2) \rightsquigarrow 4 \times (2 \times 2) \rightsquigarrow 4 \times 4 \rightsquigarrow 16$$

What become those ideas when applied to λ -calculus ?

- ▶ In λ -calculus, **sense transformation** is realized by β -reduction
- ▶ **An existentialist view : action comes first** (we do not know a priori the set of denotations “behind” the terms, if any)
- ▶ To (try to) make the set of denotations emerge from the dynamic, one needs (to start with) to equalize any two senses such that one is converted into the other one by β -reduction: that is the concept of **β -equivalence** to be defined now.

β -équivalence

Definition

- ▶ The relation \sim_β (**β -équivalence**) is the smallest equivalence relation on Λ in which $\overset{\beta}{\rightsquigarrow}_1$ is included. **Conceptually: \sim_β is just the equality over terms/senses forced by the computation.**

β -équivalence

Definition

- ▶ The relation \sim_β (**β -équivalence**) is the smallest equivalence relation on Λ in which $\overset{\beta}{\rightsquigarrow}_1$ is included. **Conceptually: \sim_β is just the equality over terms/senses forced by the computation.**
- ▶ The β -quotient (Λ / \sim_β) : morally, **an equivalence class for \sim_β = a set of terms/senses having a same denotation.**

β -équivalence

Definition

- ▶ The relation \sim_β (**β -équivalence**) is the smallest equivalence relation on Λ in which $\overset{\beta}{\rightsquigarrow}_1$ is included. **Conceptually: \sim_β is just the equality over terms/senses forced by the computation.**
- ▶ The β -quotient (Λ / \sim_β): morally, **an equivalence class for \sim_β = a set of terms/senses having a same denotation.**

But equalization is contagious: equalizing two terms forces to equalize other ones (if Perceval = Adrien, then Perceval's sister = Adrien's sister etc). Risk of a semantical collapse: only one equivalence class (only one denotation, only one individual).

β -équivalence

Definition

- ▶ The relation \sim_β (**β -équivalence**) is the smallest equivalence relation on Λ in which $\overset{\beta}{\rightsquigarrow}_1$ is included. **Conceptually: \sim_β is just the equality over terms/senses forced by the computation.**
- ▶ The β -quotient (Λ / \sim_β): morally, **an equivalence class for \sim_β = a set of terms/senses having a same denotation.**

But equalization is contagious: equalizing two terms forces to equalize other ones (if Perceval = Adrien, then Perceval's sister = Adrien's sister etc). Risk of a semantical collapse: only one equivalence class (only one denotation, only one individual).

Computational (semantical) consistency

- ▶ If two normalizable terms t, u are β -equivalent, then they have the same normal form (a corollary of β -reduction confluence)
- ▶ Corollary : **the variety of normal forms survives to the \sim_β -equalization (consistency)**

Is β -equivalence enough?

Back to the idea that our individuals are acting ones
(remember: “tell me how t acts, I will say you who t is”).

Question: if t and u are not β -equalized, does it mean that they indeed behave/act differently over at least one context/argument?

Is β -equivalence enough?

Back to the idea that our individuals are acting ones (remember: “tell me how t acts, I will say you who t is”).

Question: if t and u are not β -equalized, does it mean that they indeed behave/act differently over at least one context/argument?

Actional behavior of terms can be considered at various scales of refinement/abstraction (depending on which information about the process one accepts to forget or want to focus on). And the answer will depend on the chosen scale.

Is β -equivalence enough?

Back to the idea that our individuals are acting ones (remember: “tell me how t acts, I will say you who t is”).

Question: if t and u are not β -equalized, does it means that they indeed behave/act differently over at least one context/argument?

Actional behavior of terms can be considered at various scales of refinement/abstraction (depending on which information about the process one accepts to forget or want to focus on). And the answer will depend on the chosen scale.

Let us start at the coarser-grained scale (the “**extensional scale**”), i.e. the one where we forgets almost everything of the actional processes induced by a term t , keeping only the:

- ▶ starting points (the arguments, the **inputs**)
- ▶ and the arrival points (the results of the corresponding computation, the **outputs**).

The η -equivalence and the $\beta\eta$ quotient

Our question (recalled): if t and u are not β -equalized, does it mean that they indeed behave/act differently over at least one context/argument?

At the “extensional scale”, the answer is: **no**. Indeed the two following non β -equivalent terms:

1. t
2. $\lambda x (t x)$ (where x is not free in t)

act identically over any argument u , as $(\lambda x (t x) u) \rightsquigarrow_{\beta_1} (t u)$

The η -equivalence and the $\beta\eta$ quotient

Our question (recalled): if t and u are not β -equalized, does it mean that they indeed behave/act differently over at least one context/argument?

At the “extensional scale”, the answer is: **no**. Indeed the two following non β -equivalent terms:

1. t
2. $\lambda x (t x)$ (where x is not free in t)

act identically over any argument u , as $(\lambda x (t x) u) \rightsquigarrow_{\beta_1} (t u)$

So, we need to extend β -equivalence, in order to equalize also terms which differ with that respect (but are else extensionally indiscernible):

The η -equivalence and the $\beta\eta$ quotient

Our question (recalled): if t and u are not β -equalized, does it mean that they indeed behave/act differently over at least one context/argument?

At the “extensional scale”, the answer is: **no**. Indeed the two following non β -equivalent terms:

1. t
2. $\lambda x (t x)$ (where x is not free in t)

act identically over any argument u , as $(\lambda x (t x) u) \rightsquigarrow_{\beta_1} (t u)$

So, we need to extend β -equivalence, in order to equalize also terms which differ with that respect (but are else extensionally indiscernible):

Definitions

- ▶ η -equivalence: $\dots \lambda x (t x) \dots \approx_{\eta} \dots t \dots$ (where $x \notin t$)
- ▶ $\beta\eta$ -equivalence and the $\beta\eta$ -quotient

The η -equivalence and the $\beta\eta$ quotient

Our question (recalled): if t and u are not β -equalized, does it mean that they indeed behave/act differently over at least one context/argument?

At the “**extensional scale**”, the answer is: **no**. Indeed **the two following non β -equivalent terms:**

1. t
2. $\lambda x(t x)$ (where x is not free in t)

act identically over any argument u , as $(\lambda x(t x) u) \rightsquigarrow_{\beta_1} (t u)$

So, we need to extend β -equivalence, in order to equalize also terms which differ with that respect (but are else extensionally indiscernible):

Definitions

- ▶ η -equivalence: $\dots \lambda x(t x) \dots \approx_{\eta} \dots t \dots$ (where $x \notin t$)
- ▶ $\beta\eta$ -equivalence and the $\beta\eta$ -quotient

Reformulated for $\approx_{\beta\eta}$, our question now receives a positive answer: **if t and u are not $\beta\eta$ -equalized, then there exists a context over which they behave/act differently**

Toward recovering extensions

Toward recovering the extensional semantical viewpoint from the actional semantical viewpoint. Means to **uncover the functions (in the extensional sense) that a term t actionally induces.**

Toward recovering extensions

Toward recovering the extensional semantical viewpoint from the actional semantical viewpoint. Means to **uncover the functions (in the extensional sense) that a term t actionally induces.**

The set of closed normal terms plays for us the role of **a screen** to grasp extensional extensions: **inputs** and **outputs**.

Toward recovering extensions

Toward recovering the extensional semantical viewpoint from the actional semantical viewpoint. Means to **uncover the functions (in the extensional sense) that a term t actionally induces.**

The set of closed normal terms plays for us the role of **a screen** to grasp extensional extensions: **inputs** and **outputs**.

One focuses on a subset whose elements will “represent” natural integers, namely the subset of **Church’s integers**:

$$\underline{n} = \lambda x \lambda f \overbrace{(f \dots (f x) \dots)}^{n \text{ times}}$$

For a given $n \in \mathbb{N}$, actionally speaking, **the term \underline{n} is “the iterator” n times”**.

A philosophical belvedere for the actional semantics viewpoint: a beautiful actional, algorithmical account of integers

Recovering extensions

Church's integers

$$\underline{n} = \lambda x \lambda f \overbrace{(f \dots (f x) \dots)}^{n \text{ times}}$$

The set $\{\underline{n}\}_{n \in \mathbb{N}}$ of the λ -terms representing integers is ready to play the role of an "extensional screen" required to ask: which functions from integers to integers are realized?

Recovering extensions

Church's integers

$$\underline{n} = \lambda x \lambda f \overbrace{(f \dots (f x) \dots)}^{n \text{ times}}$$

The set $\{\underline{n}\}_{n \in \mathbb{N}}$ of the λ -terms representing integers is ready to play the role of an "extensional screen" required to ask: which functions from integers to integers are realized?

Definition: Function (over integers) "represented" by a λ -term

A k -ary function $F : \mathbb{N}^k \rightarrow \mathbb{N}$ is "represented" by a term t acting on Church's integers, if for any k -tuple (n_1, \dots, n_k) of integers:

- ▶ if $F(n_1 \dots n_k)$ is defined, then :

$$(t \underline{n_1} \dots \underline{n_k}) \rightsquigarrow_{\beta} \underline{F(n_1 \dots n_k)}$$

- ▶ Else:

$$(t \underline{n_1} \dots \underline{n_k}) \text{ is not normalizable,}$$

λ -calculus: a Turing complete programming language

Partial recursive functions : defined by a couple of basis functions over integers and two closure schemes (composition and mu-scheme).

Church's representation theorem. λ -calculus is Turing complete:
Any partial recursive function is represented by some λ -term over Church's integers (the basis functions and the closure schemes can be represented in λ -calculus).

λ -calculus: a Turing complete programming language

Partial recursive functions : defined by a couple of basis functions over integers and two closure schemes (composition and mu-scheme).

Church's representation theorem. λ -calculus is Turing complete:
Any partial recursive function is represented by some λ -term over Church's integers (the basis functions and the closure schemes can be represented in λ -calculus).

NB : complexity \Leftrightarrow duplications \Leftrightarrow multiple occurrence of a same variable

λ -calculus: a Turing complete programming language

Partial recursive functions : defined by a couple of basis functions over integers and two closure schemes (composition and mu-scheme).

Church's representation theorem. λ -calculus is Turing complete:
Any partial recursive function is represented by some λ -term over Church's integers (the basis functions and the closure schemes can be represented in λ -calculus).

NB : complexity \Leftrightarrow duplications \Leftrightarrow multiple occurrence of a same variable

What about functionals ?

- ▶ Scott Domains (a denotational semantics for Lambda-Calculus)
- ▶ Girard's decomposition : coherent spaces

PART 2

PROOFS IN CLASSICAL LOGIC

Important steps in the history of Proofs representation

David Hilbert's Proof theory program

- ▶ Prove the consistency of formal methods (Peano axioms for arithmetics)
- ▶ Through a new branch of mathematics : “Proof theory”, studying mathematically mathematical proofs

Important steps in the history of Proofs representation

David Hilbert's Proof theory program

- ▶ Prove the consistency of formal methods (Peano axioms for arithmetics)
- ▶ Through a new branch of mathematics : “Proof theory”, studying mathematically mathematical proofs
- ▶ What is a proof ?

Important steps in the history of Proofs representation

David Hilbert's Proof theory program

- ▶ Prove the consistency of formal methods (Peano axioms for arithmetics)
- ▶ Through a new branch of mathematics : “Proof theory”, studying mathematically mathematical proofs
- ▶ What is a proof ?
 - ▶ Hilbertian answer : a discourse respecting the rules of logic (local correctness)

Important steps in the history of Proofs representation

David Hilbert's Proof theory program

- ▶ Prove the consistency of formal methods (Peano axioms for arithmetics)
- ▶ Through a new branch of mathematics : “Proof theory”, studying mathematically mathematical proofs
- ▶ What is a proof ?
 - ▶ Hilbertian answer : a discourse respecting the rules of logic (local correctness)
 - ▶ Other possible answers : e.g. a universal strategy against any argumentative attack (dialectic answer)

Important steps in the history of Proofs representation

- ▶ Proofs as “texts” (written in the ideographical language of formal logic)

Important steps in the history of Proofs representation

- ▶ Proofs as “texts” (written in the ideographical language of formal logic)
 - ▶ D. Hilbert’s proof systems

Important steps in the history of Proofs representation

- ▶ Proofs as “texts” (written in the ideographical language of formal logic)
 - ▶ D. Hilbert’s proof systems
 - ▶ Natural Deduction, G. Gentzen, +/-1930

Important steps in the history of Proofs representation

- ▶ Proofs as “texts” (written in the ideographical language of formal logic)
 - ▶ D. Hilbert’s proof systems
 - ▶ Natural Deduction, G. Gentzen, +/-1930
 - ▶ Sequent calculus

Important steps in the history of Proofs representation

- ▶ Proofs as “texts” (written in the ideographical language of formal logic)
 - ▶ D. Hilbert’s proof systems
 - ▶ Natural Deduction, G. Gentzen, +/-1930
 - ▶ Sequent calculus
- ▶ As non textual objects: JY. Girard’s Proof-nets

Important steps in the history of Proofs representation

- ▶ Proofs as “texts” (written in the ideographical language of formal logic)
 - ▶ D. Hilbert’s proof systems
 - ▶ Natural Deduction, G. Gentzen, +/-1930
 - ▶ Sequent calculus
- ▶ As non textual objects: JY. Girard’s Proof-nets
 - ▶ Spatial 3-dimensional objects

Important steps in the history of Proofs representation

- ▶ Proofs as “texts” (written in the ideographical language of formal logic)
 - ▶ D. Hilbert’s proof systems
 - ▶ Natural Deduction, G. Gentzen, +/-1930
 - ▶ Sequent calculus
- ▶ As non textual objects: JY. Girard’s Proof-nets
 - ▶ Spatial 3-dimensional objects
 - ▶ non sequential

Important steps in the history of Proofs representation

- ▶ Proofs as “texts” (written in the ideographical language of formal logic)
 - ▶ D. Hilbert’s proof systems
 - ▶ Natural Deduction, G. Gentzen, +/-1930
 - ▶ Sequent calculus
- ▶ As non textual objects: JY. Girard’s Proof-nets
 - ▶ Spatial 3-dimensional objects
 - ▶ non sequential
 - ▶ global correctness

Important steps in the history of Proofs representation

- ▶ Proofs as “texts” (written in the ideographical language of formal logic)
 - ▶ D. Hilbert’s proof systems
 - ▶ Natural Deduction, G. Gentzen, +/-1930
 - ▶ Sequent calculus
- ▶ As non textual objects: JY. Girard’s Proof-nets
 - ▶ Spatial 3-dimensional objects
 - ▶ non sequential
 - ▶ global correctness

Let us start with Natural Deduction :

- ▶ Proofs as texts,
- ▶ deriving statements (formulas) from statements (formulas)

Propositional formulas

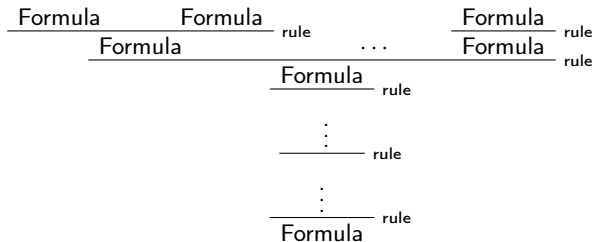
Example (a formula):

$$((\neg X \rightarrow Y) \wedge X) \vee \neg(Y \rightarrow \perp)$$

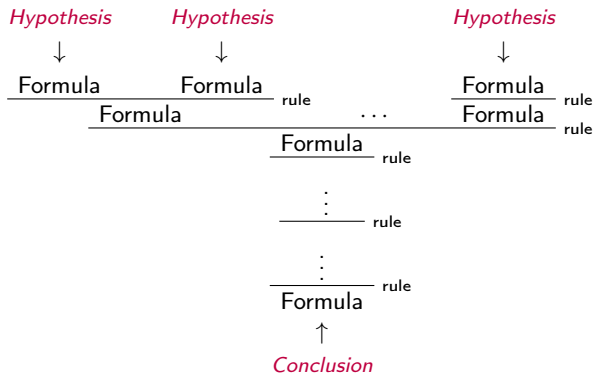
Formulas $A, B, C \dots$ are inductively built :

- ▶ from elementary formulas :
 - ▶ **atoms** : X, Y, Z etc (atomic formulas)
 - ▶ and **absurdum** : \perp
- ▶ by applying :
 - ▶ the unary constructor **negation** : $\neg A$
 - ▶ the binary constructors :
 - ▶ **conjunction**: $A \wedge B$
 - ▶ **disjunction**: $A \vee B$

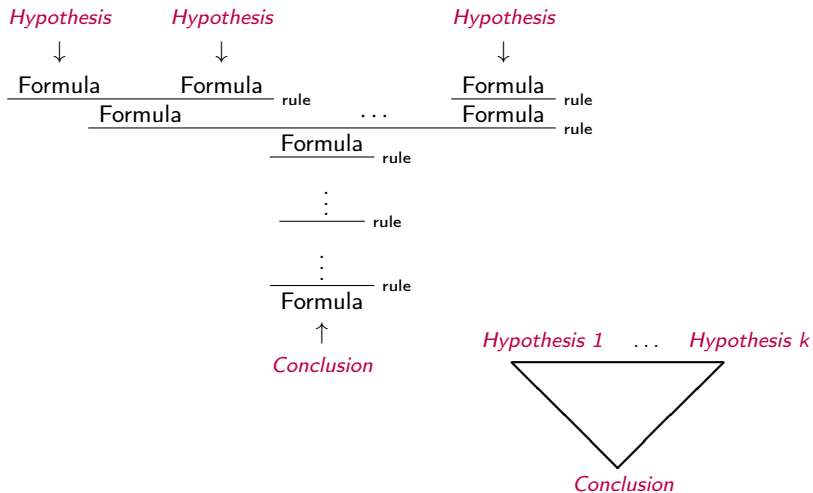
Natural Deduction: General shape of proofs (tree)



Natural Deduction: General shape of proofs (tree)



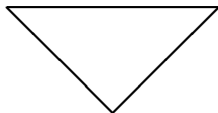
Natural Deduction: General shape of proofs (tree)



Proofs in Natural Deduction : an example

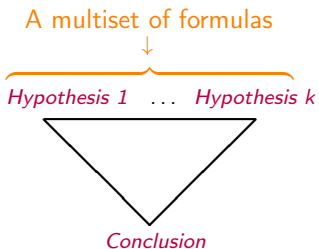
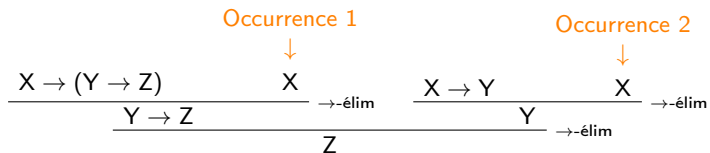
$$\frac{\frac{X \rightarrow (Y \rightarrow Z)}{Y \rightarrow Z} \quad X}{Z} \rightarrow\text{-}\acute{\text{e}}\text{lim} \quad \frac{\frac{X \rightarrow Y}{Y} \quad X}{X} \rightarrow\text{-}\acute{\text{e}}\text{lim}$$

Hypothesis 1 ... Hypothesis k

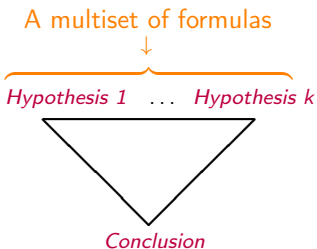
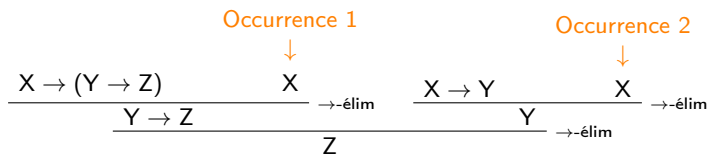


Conclusion

Proofs in Natural Deduction : an example



Proofs in Natural Deduction : an example

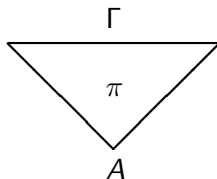


A **multiset** = a set "with repetitions" (no matter the order)

Representation of proofs in Natural Deduction

Notations:

- ▶ Formulas : A, B, C etc
- ▶ Multisets of formulas: Γ, Δ etc
- ▶ Proofs: π etc



Simply represented by

Γ
 $\vdots \pi$
 A

Rules

Introductions	Eliminations
$\frac{A \quad B}{A \wedge B} \wedge\text{-intro}$	$\frac{A \wedge B}{A} \wedge\text{-elim (1)} \quad \frac{A \wedge B}{B} \wedge\text{-elim (2)}$
	$\frac{A \rightarrow B \quad A}{B} \rightarrow\text{-elim}$
$\frac{A}{A \vee B} \vee\text{-intro (1)} \quad \frac{B}{A \vee B} \vee\text{-intro (2)}$	

Rules

Introductions	Eliminations
$\frac{A \quad B}{A \wedge B} \wedge\text{-intro}$	$\frac{A \wedge B}{A} \wedge\text{-elim (1)} \quad \frac{A \wedge B}{B} \wedge\text{-elim (2)}$
	$\frac{A \rightarrow B \quad A}{B} \rightarrow\text{-elim}$
$\frac{A}{A \vee B} \vee\text{-intro (1)} \quad \frac{B}{A \vee B} \vee\text{-intro (2)}$	

Remark. Rules are constructors for proofs (not transitions from formulas to formulas)

Rules

Introductions	Eliminations
$\frac{\begin{array}{c} \Gamma \\ \vdots \\ A \end{array} \quad \begin{array}{c} \Gamma \\ \vdots \\ B \end{array}}{A \wedge B} \wedge\text{-intro}$	$\frac{\begin{array}{c} \Gamma \\ \vdots \\ A \wedge B \end{array}}{A} \wedge\text{-elim (1)} \quad \frac{\begin{array}{c} \Gamma \\ \vdots \\ A \wedge B \end{array}}{B} \wedge\text{-elim (2)}$
	$\frac{\begin{array}{c} \Gamma \\ \vdots \\ A \rightarrow B \end{array} \quad \begin{array}{c} \Gamma \\ \vdots \\ A \end{array}}{B} \rightarrow\text{-elim}$
$\frac{\begin{array}{c} \Gamma \\ \vdots \\ A \end{array}}{A \vee B} \vee\text{-intro (1)} \quad \frac{\begin{array}{c} \Gamma \\ \vdots \\ B \end{array}}{A \vee B} \vee\text{-intro (2)}$	

Remark. Rules are constructors for proofs (not transitions from formulas to formulas)

Logical rules actually are proofs constructors

For instance, the rule \wedge -elim₁ presented as :

$$\frac{\begin{array}{c} \Gamma \\ \vdots \\ A \wedge B \end{array}}{A} \wedge\text{-élim}_1$$

Logical rules actually are proofs constructors

For instance, the rule \wedge -elim₁ presented as :

$$\frac{\begin{array}{c} \Gamma \\ \vdots \\ A \wedge B \end{array}}{A} \wedge\text{-elim}_1$$

is just a shortcut for :

$$\begin{array}{c} \Gamma \\ \vdots \pi \\ A \wedge B \end{array} \quad \mapsto \quad \left. \begin{array}{c} \Gamma \\ \vdots \pi \\ \frac{A \wedge B}{A} \wedge\text{-elim}_1 \end{array} \right\} \pi'$$

Proof π
constructed
at Time t

New proof π'
constructed
at Time $t + 1$

Rules

Introductions	Eliminations
$\frac{\begin{array}{c} \Gamma \\ \vdots \\ \pi_1 \\ A \end{array} \quad \begin{array}{c} \Gamma \\ \vdots \\ \pi_2 \\ B \end{array}}{A \wedge B} \wedge\text{-intro}$	$\frac{\begin{array}{c} \Gamma \\ \vdots \\ \pi \\ A \wedge B \end{array}}{A} \wedge\text{-elim (1)} \quad \frac{\begin{array}{c} \Gamma \\ \vdots \\ \pi \\ A \wedge B \end{array}}{B} \wedge\text{-elim (2)}$
	$\frac{\begin{array}{c} \Gamma \\ \vdots \\ \pi_1 \\ A \rightarrow B \end{array} \quad \begin{array}{c} \Gamma \\ \vdots \\ \pi_2 \\ A \end{array}}{B} \rightarrow\text{-elim}$
$\frac{\begin{array}{c} \Gamma \\ \vdots \\ \pi \\ A \end{array}}{A \vee B} \vee\text{-intro (1)} \quad \frac{\begin{array}{c} \Gamma \\ \vdots \\ \pi_3 \\ B \end{array}}{A \vee B} \vee\text{-intro (2)}$	

Rules

Introductions	Eliminations
$\frac{\begin{array}{c} \Gamma \\ \vdots \\ \pi_1 \\ A \end{array} \quad \begin{array}{c} \Gamma \\ \vdots \\ \pi_2 \\ B \end{array}}{A \wedge B} \wedge\text{-intro}$	$\frac{\begin{array}{c} \Gamma \\ \vdots \\ \pi \\ A \wedge B \end{array}}{A} \wedge\text{-elim (1)} \quad \frac{\begin{array}{c} \Gamma \\ \vdots \\ \pi \\ A \wedge B \end{array}}{B} \wedge\text{-elim (2)}$
$\frac{\begin{array}{c} \Gamma \\ \vdots \\ \pi \\ B \end{array}}{A \rightarrow B} \rightarrow\text{-intro}$	$\frac{\begin{array}{c} \Gamma \\ \vdots \\ \pi_1 \\ A \rightarrow B \end{array} \quad \begin{array}{c} \Gamma \\ \vdots \\ \pi_2 \\ A \end{array}}{B} \rightarrow\text{-elim}$
$\frac{\begin{array}{c} \Gamma \\ \vdots \\ \pi \\ A \end{array}}{A \vee B} \vee\text{-intro (1)} \quad \frac{\begin{array}{c} \Gamma \\ \vdots \\ \pi_3 \\ B \end{array}}{A \vee B} \vee\text{-intro (2)}$	$\frac{\begin{array}{c} \Gamma \\ \vdots \\ \pi_1 \\ A \vee B \end{array} \quad \begin{array}{c} \Gamma \\ \vdots \\ \pi_2 \\ C \end{array} \quad \begin{array}{c} \Gamma \\ \vdots \\ \pi_3 \\ C \end{array}}{C} \vee\text{-elim}$

Rules

Introductions	Eliminations
$\begin{array}{c} \Gamma \ A^n \\ \vdots \ \pi \\ n \ \frac{B}{A \rightarrow B} \rightarrow\text{-intro} \end{array}$	
	$\begin{array}{ccc} \Gamma & \Gamma \ A^n & \Gamma \ B^n \\ \vdots \ \pi_1 & \vdots \ \pi_2 & \vdots \ \pi_3 \\ n \ \frac{A \vee B}{C} & C & C \quad \vee\text{-elim} \end{array}$

Rules

Introductions	Eliminations
$\begin{array}{c} \Gamma \quad A^n \\ \vdots \quad \pi \\ n \frac{B}{A \rightarrow B} \rightarrow\text{-intro} \end{array}$	

Remark 3. Hypothesis desactivation

Rules

Hypothesis
“active”
at Time $t \dots$

\dots are
desactivated
at Time $t + 1$

$$\Gamma \quad \left. \begin{array}{c} \text{k times} \\ \overbrace{A \dots A} \\ \vdots \pi \\ B \end{array} \right\}$$

\mapsto

$$\Gamma \quad \left. \begin{array}{c} \text{k times} \\ \overbrace{A^n \dots A^n} \\ \vdots \pi \\ \begin{array}{c} B \\ \hline A \rightarrow B \end{array} \rightarrow\text{-intro} \end{array} \right\} \pi'$$

Proof π
constructed
at Time t

New proof π'
constructed
at Time $t + 1$

A proof in Natural Deduction : an example

$$\frac{\frac{\frac{\cancel{X \rightarrow (Y \rightarrow Z)}^3 \quad \cancel{X}^1}{Y \rightarrow Z} \rightarrow\text{-élim} \quad \frac{\frac{\cancel{X \rightarrow Y}^2 \quad \cancel{X}^1}{Y} \rightarrow\text{-élim}}{Z} \rightarrow\text{-intro}}{\frac{X \rightarrow Y \rightarrow (X \rightarrow Z)}{(X \rightarrow Y) \rightarrow (X \rightarrow Z)} \rightarrow\text{-intro}} \rightarrow\text{-intro}$$

3 $\frac{(X \rightarrow (Y \rightarrow Z)) \rightarrow ((X \rightarrow Y) \rightarrow (X \rightarrow Z))}{(X \rightarrow (Y \rightarrow Z)) \rightarrow ((X \rightarrow Y) \rightarrow (X \rightarrow Z))} \rightarrow\text{-intro}$

PART 2

The computational point of view on proofs

An example of non analytical proof

$$\frac{\frac{\frac{X^2}{X \wedge Y} \wedge\text{-intro}}{\frac{X \wedge Y}{X} \wedge\text{-elim (1)}}}{\frac{1 \frac{Y \rightarrow X}{Y \rightarrow X} \rightarrow\text{-intro}}{2 \frac{X \rightarrow (Y \rightarrow X)}{X \rightarrow (Y \rightarrow X)} \rightarrow\text{-intro}}}$$

An example of non analytical proof

$$\frac{\frac{\frac{X^2}{X} \quad \frac{Y^1}{Y}}{X \wedge Y} \wedge\text{-intro}}{X} \wedge\text{-elim (1)}}{1 \quad \frac{Y \rightarrow X}{Y \rightarrow X} \rightarrow\text{-intro}} \rightarrow\text{-intro} \\ 2 \quad \frac{Y \rightarrow X}{X \rightarrow (Y \rightarrow X)} \rightarrow\text{-intro}$$

- ▶ The connective \wedge is not present in $X \rightarrow (Y \rightarrow X)$ (the proved theorem)

An example of non analytical proof

$$\frac{\frac{\frac{X^2}{X} \quad \frac{Y^1}{Y}}{X \wedge Y} \wedge\text{-intro}}{X} \wedge\text{-elim (1)}}{1 \quad \frac{Y \rightarrow X}{Y \rightarrow X} \rightarrow\text{-intro}} \rightarrow\text{-intro} \\ 2 \quad \frac{Y \rightarrow X}{X \rightarrow (Y \rightarrow X)} \rightarrow\text{-intro}$$

- ▶ The connective \wedge is not present in $X \rightarrow (Y \rightarrow X)$ (the proved theorem)
- ▶ \wedge is an extrinsic element : it cannot be obtained by analyzing the theorem (nor the hypothesis)

An example of non analytical proof

$$\frac{\frac{\frac{X^2}{X} \quad \frac{Y^1}{Y}}{X \wedge Y} \wedge\text{-intro}}{X} \wedge\text{-elim (1)}}{1 \quad \frac{Y \rightarrow X}{Y \rightarrow X} \rightarrow\text{-intro}} \rightarrow\text{-intro} \\ 2 \quad \frac{Y \rightarrow X}{X \rightarrow (Y \rightarrow X)} \rightarrow\text{-intro}$$

- ▶ The connective \wedge is not present in $X \rightarrow (Y \rightarrow X)$ (the proved theorem)
- ▶ \wedge is an extrinsic element : it cannot be obtained by analyzing the theorem (nor the hypothesis)
- ▶ The proof is **not analytical** (it contains an extrinsic element)

An example of non analytical proof

$$\frac{\frac{\frac{X^2}{X} \quad \frac{Y^1}{Y}}{X \wedge Y} \wedge\text{-intro}}{X} \wedge\text{-elim (1)}}{1 \quad \frac{Y \rightarrow X}{Y \rightarrow X} \rightarrow\text{-intro}} \rightarrow\text{-intro} \\ 2 \quad \frac{X \rightarrow (Y \rightarrow X)}{X \rightarrow (Y \rightarrow X)} \rightarrow\text{-intro}$$

- ▶ The connective \wedge is not present in $X \rightarrow (Y \rightarrow X)$ (the proved theorem)
- ▶ \wedge is an extrinsic element : it cannot be obtained by analyzing the theorem (nor the hypothesis)
- ▶ The proof is **not analytical** (it contains an extrinsic element)

Importance of analyticity, w.r.t. heuristics (proof-search):

knowing the statement we want to prove, we know in advance the **finite** list of formulas that could appear in the proof we are looking for (namely the subformulas of the statement supposed to be proved).

Back to our example of non analytical proof

$$\frac{\frac{\frac{X^2}{X \wedge Y} \wedge\text{-intro}}{\frac{X}{Y \rightarrow X}} \wedge\text{-elim (1)}}{X \rightarrow (Y \rightarrow X)} \rightarrow\text{-intro}$$

1 $\frac{Y \rightarrow X}{Y \rightarrow X} \rightarrow\text{-intro}$

2 $\frac{X \rightarrow (Y \rightarrow X)}{X \rightarrow (Y \rightarrow X)} \rightarrow\text{-intro}$

Back to our example of non analytical proof

$$\frac{\frac{\frac{X^2}{X \wedge Y} \wedge\text{-intro}}{\frac{X}{Y \rightarrow X}} \wedge\text{-elim (1)} \rightarrow\text{-intro}}{X \rightarrow (Y \rightarrow X)} \rightarrow\text{-intro}$$

1

2

- ▶ In that proof : scarcely introduced, the extrinsic element \wedge happens to be eliminated

Back to our example of non analytical proof

$$\left. \begin{array}{c} \frac{X^2 \quad Y^1}{X \wedge Y} \wedge\text{-intro} \\ \frac{X \wedge Y}{X} \wedge\text{-elim (1)} \\ 1 \frac{X}{Y \rightarrow X} \rightarrow\text{-intro} \\ 2 \frac{Y \rightarrow X}{X \rightarrow (Y \rightarrow X)} \rightarrow\text{-intro} \end{array} \right\} \text{ a "cut" (alias a "redex")}$$

- ▶ In that proof : scarcely introduced, the extrinsic element \wedge happens to be eliminated

Back to our example of non analytical proof

$$\frac{\frac{\frac{X^2}{X} \quad \frac{Y^1}{Y}}{X \wedge Y} \wedge\text{-intro}}{X} \wedge\text{-elim (1)}}{1 \frac{Y \rightarrow X}{Y \rightarrow X} \rightarrow\text{-intro}} \rightarrow\text{-intro}$$
$$2 \frac{Y \rightarrow X}{X \rightarrow (Y \rightarrow X)} \rightarrow\text{-intro}$$

- ▶ In that proof : scarcely introduced, the extrinsic element \wedge happens to be eliminated
- ▶ Gentzen proved:
 - ▶ The situation above is general: **Proofs with no cut are analytical** (NB : wrong in second order logic)

Back to our example of non analytical proof

$$\frac{\frac{\frac{X^2}{X} \quad \frac{Y^1}{Y}}{X \wedge Y} \wedge\text{-intro}}{X} \wedge\text{-elim (1)}}{\frac{1 \quad \frac{Y \rightarrow X}{X}}{Y \rightarrow X} \rightarrow\text{-intro}}{X \rightarrow (Y \rightarrow X)} \rightarrow\text{-intro}$$

- ▶ In that proof : scarcely introduced, the extrinsic element \wedge happens to be eliminated
- ▶ Gentzen proved:
 - ▶ The situation above is general: **Proofs with no cut are analytical** (NB : wrong in second order logic)
 - ▶ **Analytizability of proofs** : one can transform any proof, in a cut-free (hence analytical) proof of the same theorem.

Gentzen's algorithm for cut elimination: one step

$$\begin{array}{c} \Gamma_1 \quad \cancel{A^{(0)}}^n \quad \dots \quad \cancel{A^{(k)}}^n \\ \vdots \pi_1 \\ \frac{n \quad \frac{A \rightarrow B}{B} \rightarrow\text{-intro}}{A \rightarrow B} \quad \Gamma_2 \\ \frac{\quad \frac{A}{A} \rightarrow\text{-elim}}{A} \rightarrow\text{-elim} \\ \hline B \\ \vdots \end{array}$$

Gentzen's algorithm for cut elimination: one step

$$\begin{array}{c}
 \Gamma_1 \quad \cancel{A^{(0)}}^n \quad \dots \quad \cancel{A^{(k)}}^n \\
 \vdots \pi_1 \\
 n \frac{B}{A \rightarrow B} \xrightarrow{\text{intro}} \\
 \hline
 B \\
 \vdots
 \end{array}
 \quad
 \begin{array}{c}
 \Gamma_2 \\
 \vdots \pi_2 \\
 A \xrightarrow{\text{elim}}
 \end{array}
 \quad
 \rightsquigarrow
 \quad
 \begin{array}{c}
 \Gamma_2^{(0)} \quad \Gamma_2^{(k)} \\
 \vdots \pi_2^{(0)} \quad \vdots \pi_2^{(k)} \\
 \Gamma_1 \quad A^{(0)} \quad \dots \quad A^{(n)} \\
 \vdots \pi_1 [\pi_2 / A] \\
 B \\
 \vdots
 \end{array}$$

Gentzen's algorithm for cut elimination: one step

$$\begin{array}{c}
 \Gamma_1 \quad \cancel{A^{(0)}}^n \quad \dots \quad \cancel{A^{(k)}}^n \\
 \vdots \pi_1 \\
 n \frac{B}{A \rightarrow B} \xrightarrow{\text{intro}} \\
 \hline
 B \\
 \vdots
 \end{array}
 \quad
 \begin{array}{c}
 \Gamma_2 \\
 \vdots \pi_2 \\
 A \xrightarrow{\text{elim}}
 \end{array}
 \quad
 \rightsquigarrow
 \quad
 \begin{array}{c}
 \Gamma_2^{(0)} \quad \Gamma_2^{(k)} \\
 \vdots \pi_2^{(0)} \quad \vdots \pi_2^{(k)} \\
 \Gamma_1 \quad A^{(0)} \quad \dots \quad A^{(n)} \\
 \vdots \pi_1 [\pi_2 / A] \\
 B \\
 \vdots
 \end{array}$$

- Complexity \Leftrightarrow Duplication \Leftrightarrow Multiple occurrences hypothesis

Why is cut-eliminability so important ?

- ▶ **Logical reasons:** Non analytical proofs

Why is cut-eliminability so important ?

- ▶ **Logical reasons:** Non analytical proofs
 - ▶ are epistemologically dubious : non intrinsic motivation of the theorem

Why is cut-eliminability so important ?

- ▶ **Logical reasons:** Non analytical proofs
 - ▶ are epistemologically dubious : non intrinsic motivation of the theorem
 - ▶ give no bound on the number of different formulas to consider in the heuristical (proof-search) process

Why is cut-eliminability so important ?

- ▶ **Logical reasons:** Non analytical proofs
 - ▶ are epistemologically dubious : non intrinsic motivation of the theorem
 - ▶ give no bound on the number of different formulas to consider in the heuristical (proof-search) process

A (potential) corollary of cut-eliminability : Consistency results

Why is cut-eliminability so important ?

- ▶ **Logical reasons:** Non analytical proofs
 - ▶ are epistemologically dubious : non intrinsic motivation of the theorem
 - ▶ give no bound on the number of different formulas to consider in the heuristical (proof-search) process

A (potential) corollary of cut-eliminability : Consistency results

For Proof theory cut-elimination is the corner stone of Logic.

Why is cut-eliminability so important ?

- ▶ **Logical reasons:** Non analytical proofs
 - ▶ are epistemologically dubious : non intrinsic motivation of the theorem
 - ▶ give no bound on the number of different formulas to consider in the heuristical (proof-search) process

A (potential) corollary of cut-eliminability : Consistency results

For Proof theory cut-elimination is the corner stone of Logic.

- ▶ **Computing theory reasons:**

Why is cut-eliminability so important ?

- ▶ **Logical reasons:** Non analytical proofs
 - ▶ are epistemologically dubious : non intrinsic motivation of the theorem
 - ▶ give no bound on the number of different formulas to consider in the heuristical (proof-search) process

A (potential) corollary of cut-eliminability : Consistency results

For Proof theory cut-elimination is the corner stone of Logic.

- ▶ **Computing theory reasons:**
 - ▶ Lambda calculus (Church, 1934) : a functional ontology; a programming language

Why is cut-eliminability so important ?

- ▶ **Logical reasons:** Non analytical proofs
 - ▶ are epistemologically dubious : non intrinsic motivation of the theorem
 - ▶ give no bound on the number of different formulas to consider in the heuristical (proof-search) process

A (potential) corollary of cut-eliminability : Consistency results

For Proof theory cut-elimination is the corner stone of Logic.

- ▶ **Computing theory reasons:**
 - ▶ Lambda calculus (Church, 1934) : a functional ontology; a programming language
 - ▶ Curry-Howard isomorphism (1969) :
$$\begin{array}{c} \text{(Typed Lambda calculus, execution)} \\ = \\ \text{(Natural Deduction, cut-elimination)} \end{array}$$

Why is cut-eliminability so important ?

- ▶ **Logical reasons:** Non analytical proofs
 - ▶ are epistemologically dubious : non intrinsic motivation of the theorem
 - ▶ give no bound on the number of different formulas to consider in the heuristical (proof-search) process

A (potential) corollary of cut-eliminability : Consistency results

For Proof theory cut-elimination is the corner stone of Logic.

- ▶ **Computing theory reasons:**
 - ▶ Lambda calculus (Church, 1934) : a functional ontology; a programming language
 - ▶ Curry-Howard isomorphism (1969) :

(Typed Lambda calculus, execution)

=

(Natural Deduction, cut-elimination)

Cut-elimination = bridge between Logic and Computing theory.

Focusing on negation

Introductions	Eliminations
$\frac{\Gamma \quad \overset{\neg}{A}^n \quad \vdots \quad \pi \quad \perp}{\neg A} \neg\text{-intro}$	$\frac{\Gamma \quad \vdots \quad \pi_1 \quad \neg A \quad \Gamma \quad \vdots \quad \pi_2 \quad A}{\perp} \rightarrow\text{-elim}$
Intuitionistic absurdum (negation)	Classical absurdum (negation)
$\frac{\Gamma \quad \vdots \quad \pi \quad \perp}{A} \text{efq}$	$\frac{\Gamma \quad \overset{\neg}{A}^n \quad \vdots \quad \perp}{A} \text{raa}$

Terminology and notations for Natural Deduction systems:

- ▶ Minimal : NM
- ▶ Intuitionistic : $NJ = NM + \text{efq}$
- ▶ Classical : $NK = NJ + \text{raa}$

The “symmetries of classical logic” : dual connectives

- ▶ Notation : $A \equiv B$ if A and B are provably equivalent in NK
- ▶ De Morgan “laws” :

$$\neg\neg A \equiv A$$

$$\neg(A \wedge B) \equiv \neg A \vee \neg B \qquad \neg A \wedge \neg B \equiv \neg(A \vee B)$$

The “symmetries of classical logic” : dual connectives

- ▶ Notation : $A \equiv B$ if A and B are provably equivalent in NK
- ▶ De Morgan “laws” :

$$\neg\neg A \equiv A$$

$$\neg(A \wedge B) \equiv \neg A \vee \neg B \qquad \neg A \wedge \neg B \equiv \neg(A \vee B)$$

- ▶ “Symmetries of classical logic” at the provability level

The “symmetries of classical logic” : dual connectives

- ▶ Notation : $A \equiv B$ if A and B are provably equivalent in NK
- ▶ De Morgan “laws” :

$$\neg\neg A \equiv A$$

$$\neg(A \wedge B) \equiv \neg A \vee \neg B \qquad \neg A \wedge \neg B \equiv \neg(A \vee B)$$

- ▶ “Symmetries of classical logic” at the provability level
- ▶ However those symmetries:
 - ▶ are not visible in NK at the level of proofs
 - ▶ i.e. are almost not reflected in NK rules

The “dissymmetries of classical Natural Deduction”

Introductions	Eliminations
	$\frac{A \wedge B}{A} \wedge\text{-elim (1)} \quad \frac{A \wedge B}{B} \wedge\text{-elim (2)}$
$\frac{A}{A \vee B} \vee\text{-intro (1)} \quad \frac{B}{A \vee B} \vee\text{-intro (2)}$	
Intuitionistic absurdum (negation)	Classical absurdum (negation)

Remark **Symmetry between intros for \vee and elim for \wedge** : a track of the duality between \wedge and \vee (“De Morgan laws”).

The “dissymmetries of classical Natural Deduction”

Introductions	Eliminations
$\frac{A \quad B}{A \wedge B} \wedge\text{-intro}$	$\frac{A \wedge B}{A} \wedge\text{-elim (1)} \quad \frac{A \wedge B}{B} \wedge\text{-elim (2)}$
$\frac{A}{A \vee B} \vee\text{-intro (1)} \quad \frac{B}{A \vee B} \vee\text{-intro (2)}$	$\begin{array}{ccc} \Gamma & \Gamma \not{A}^n & \Gamma \not{B}^n \\ \vdots \pi_1 & \vdots \pi_2 & \vdots \pi_2 \\ n \frac{A \vee B}{C} & C & C \end{array} \vee\text{-elim}$
Intuitionistic absurdum (negation)	Classical absurdum (negation)

Remark **Symmetry between intros for \vee and elim for \wedge** : a track of the duality between \wedge and \vee (“De Morgan laws”).

The “dissymmetries of classical Natural Deduction”

Introductions	Eliminations
$\frac{A \quad B}{A \wedge B} \wedge\text{-intro}$	$\frac{A \wedge B}{A} \wedge\text{-elim (1)} \quad \frac{A \wedge B}{B} \wedge\text{-elim (2)}$
$\frac{A}{A \vee B} \vee\text{-intro (1)} \quad \frac{B}{A \vee B} \vee\text{-intro (2)}$	$\frac{\begin{array}{c} \Gamma \\ \vdots \\ \pi_1 \end{array} \quad \begin{array}{c} \Gamma \quad \cancel{A}^n \\ \vdots \\ \pi_2 \end{array} \quad \begin{array}{c} \Gamma \quad \cancel{B}^n \\ \vdots \\ \pi_2 \end{array}}{n \frac{A \vee B}{C} \quad C} \vee\text{-elim}$
$\frac{\begin{array}{c} \Gamma \quad \cancel{A}^n \\ \vdots \\ \pi \\ \perp \end{array}}{n \frac{\perp}{\neg A}} \neg\text{-intro}$	$\frac{\begin{array}{c} \Gamma \\ \vdots \\ \pi_1 \\ \neg A \end{array} \quad \begin{array}{c} \Gamma \\ \vdots \\ \pi_2 \\ A \end{array}}{\perp} \rightarrow\text{-elim}$
Intuitionistic absurdum (negation)	Classical absurdum (negation)
$\frac{\begin{array}{c} \Gamma \\ \vdots \\ \pi \\ \perp \end{array}}{A} \text{efq}$	$\frac{\begin{array}{c} \Gamma \quad \cancel{A}^n \\ \vdots \\ \perp \end{array}}{n \frac{\perp}{A}} \text{raa}$

Remark **Symmetry between intros for \vee and elim for \wedge** : a track of the duality between \wedge and \vee (“De Morgan laws”).

From Natural Deduction to Sequent calculus

- ▶ Two dissymmetries in the Natural Deduction format :
 - ▶ Dissymmetry Hypothesis/Conclusion (*many hypothesis vs one conclusion*)
 - ▶ Proofs are “conclusion oriented” (the grow down : toward conclusion)

From Natural Deduction to Sequent calculus

- ▶ Two dissymmetries in the Natural Deduction format :
 - ▶ Dissymmetry Hypothesis/Conclusion (*many hypothesis vs one conclusion*)
 - ▶ Proofs are “conclusion oriented” (the grow down : toward conclusion)

- ▶ Two reasons to move from Natural Deduction to Sequent calculus
 - ▶ Recover classical symmetries at the level of proofs (rules)
 - ▶ Get rid of difficulties when generalizing cut elimination to classical Natural Deduction

Toward Natural Deduction as a sequent derivation system : a comparison

An arithmetical calculation
represented via moves from
arithmetical expressions to other ones:

$$(3 \times 2) \times (5 \times 3)$$

=

$$6 \times (5 \times 3)$$

=

$$6 \times 15$$

=

$$90$$

Representation of **the same**
calculation via moves from
arithmetical identities
to other ones:

$$(3 \times 2) \times (5 \times 3) = (3 \times 2) \times (5 \times 3)$$

↓

$$(3 \times 2) \times (5 \times 3) = 6 \times (5 \times 3)$$

↓

$$(3 \times 2) \times (5 \times 3) = 6 \times 15$$

↓

$$(3 \times 2) \times (5 \times 3) = 90$$

Natural Deduction deriving sequents from sequents

Representation of the progression
of a proof via steps from
formulas to formulas:

$$\frac{\frac{\frac{X \wedge ((Y \wedge Z) \wedge W)}{(Y \wedge Z) \wedge W} \wedge\text{-elim}_1}{Y \wedge Z} \wedge\text{-elim}_2}{(Y \wedge Z) \vee X} \vee\text{-intro}_1$$

Representation of **the same**
proof via steps from
'sequents' to 'sequents':

$$\frac{\frac{\frac{X \wedge ((Y \wedge Z) \wedge W) \vdash X \wedge ((Y \wedge Z) \wedge W)}{X \wedge ((Y \wedge Z) \wedge W) \vdash (Y \wedge Z) \wedge W} \wedge\text{-elim}_1}{X \wedge ((Y \wedge Z) \wedge W) \vdash Y \wedge Z} \wedge\text{-elim}_2}{X \wedge ((Y \wedge Z) \wedge W) \vdash (Y \wedge Z) \vee X} \vee\text{-intro}_1$$

Two different *notations* for the same proof

Natural Deduction deriving sequents from sequents

Representation of the progression
of a proof via steps from
formulas to formulas:

$$\frac{\frac{\frac{X \wedge ((Y \wedge Z) \wedge W)}{(Y \wedge Z) \wedge W} \wedge\text{-elim}_1}{Y \wedge Z} \wedge\text{-elim}_2}{(Y \wedge Z) \vee X} \vee\text{-intro}_1$$

Representation of **the same**
proof via steps from
'sequents' to 'sequents':

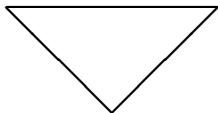
$$\frac{\frac{\frac{X \wedge ((Y \wedge Z) \wedge W) \vdash X \wedge ((Y \wedge Z) \wedge W)}{X \wedge ((Y \wedge Z) \wedge W) \vdash (Y \wedge Z) \wedge W} \wedge\text{-elim}_1}{X \wedge ((Y \wedge Z) \wedge W) \vdash Y \wedge Z} \wedge\text{-elim}_2}{X \wedge ((Y \wedge Z) \wedge W) \vdash (Y \wedge Z) \vee X} \vee\text{-intro}_1$$

Two different *notations* for the same proof

Deduction of formulas versus Derivation of sequents

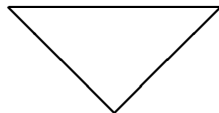
Formulas deduced from formulas

Hypothesis 1 ... *Hypothesis k*



Conclusion

Sequents derived from sequents

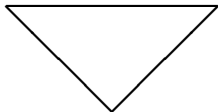


Hypothesis 1, ..., Hypothesis k \vdash *Conclusion*
A sequent

Deduction of formulas versus Derivation of sequents

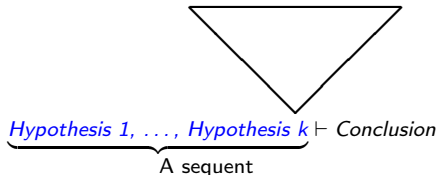
Formulas deduced from formulas

Hypothesis 1 ... *Hypothesis k*



Conclusion

Sequents derived from sequents



Γ A^n ... A^n

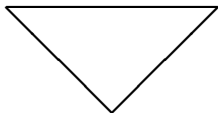
$$n \frac{\begin{array}{c} \vdots \pi \\ B \end{array}}{A \rightarrow B} \rightarrow\text{-intro}$$

$$\frac{\begin{array}{c} \vdots \\ \Gamma, A, \dots, A \vdash B \end{array}}{\Gamma \vdash A \rightarrow B} \rightarrow\text{-intro}$$

Deduction of formulas versus Derivation of sequents

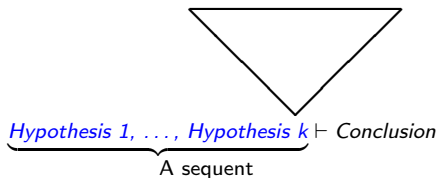
Formulas deduced from formulas

Hypothesis 1 ... *Hypothesis k*



Conclusion

Sequents derived from sequents



$\Gamma \quad A^n \dots A^n$

$\vdots \pi$

$n \frac{B}{A \rightarrow B} \rightarrow\text{-intro}$

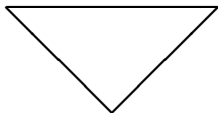
\vdots (*useless*)

$\frac{\Gamma, A, \dots, A \vdash B}{\Gamma \vdash A \rightarrow B} \rightarrow\text{-intro}$

Deduction of formulas versus Derivation of sequents

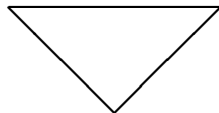
Formulas deduced from formulas

Hypothesis 1 ... *Hypothesis k*



Conclusion

Sequents derived from sequents



Hypothesis 1, ..., Hypothesis k \vdash *Conclusion*
A sequent

Γ A^n ... A^n

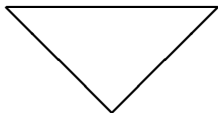
$$\begin{array}{c} \vdots \pi \\ B \\ n \frac{A \rightarrow B}{A \rightarrow B} \rightarrow\text{-intro} \end{array}$$

$$\frac{\Gamma, A, \dots, A \vdash B}{\Gamma \vdash A \rightarrow B} \rightarrow\text{-intro}$$

Deduction of formulas versus Derivation of sequents

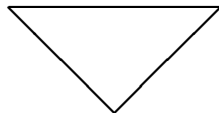
Formulas deduced from formulas

Hypothesis 1 ... *Hypothesis k*



Conclusion

Sequents derived from sequents



Hypothesis 1, ..., Hypothesis k \vdash *Conclusion*
A sequent

Hyp

↓

A

(Id ax)

↑

Concl

$\frac{}{A \vdash A}$ Id Axm

Classical Natural Deduction

Axiomes identité

$$\frac{\text{axm-id}}{\Gamma, A \vdash A}$$

$$\frac{\Gamma, A, \dots, A \vdash B}{\Gamma \vdash A \rightarrow B} \rightarrow\text{-intro}$$

$$\frac{\Gamma, A, \dots, A \vdash \perp}{\Gamma \vdash \neg A} \neg\text{-intro}$$

$$\frac{\Gamma \vdash A \quad \Gamma \vdash B}{\Gamma \vdash A \wedge B} \wedge\text{-intro}$$

$$\frac{\Gamma \vdash A}{\Gamma \vdash A \vee B} \vee\text{-i1}$$

$$\frac{\Gamma \vdash \perp}{\Gamma \vdash A} \text{efq}$$

Implication

$$\frac{\Gamma \vdash A \rightarrow B \quad \Gamma' \vdash A}{\Gamma, \Gamma' \vdash B} \rightarrow\text{-elim}$$

Négation

$$\frac{\Gamma \vdash \neg A \quad \Gamma' \vdash A}{\Gamma, \Gamma' \vdash \perp} \neg\text{-elim}$$

Conjonction

$$\frac{\Gamma \vdash A \wedge B}{\Gamma \vdash A} \wedge\text{-elim 1} \quad \frac{\Gamma \vdash A \wedge B}{\Gamma \vdash B} \wedge\text{-elim 2}$$

Disjonction

$$\frac{\Gamma \vdash B}{\Gamma \vdash A \vee B} \vee\text{-i2} \quad \frac{\Gamma \vdash A \vee B \quad \Gamma, A, \dots, A \vdash C}{\Gamma \vdash C} \vee\text{-elim}$$

L'absurde

$$\frac{\Gamma, \neg A, \dots, \neg A \vdash \perp}{\Gamma \vdash A} \text{raa}$$

Classical Natural Deduction : version 1

Axiomes identité

$$\frac{\text{axm-id}}{\Gamma, A \vdash A}$$

$$\frac{\Gamma, A, \dots, A \vdash B}{\Gamma \vdash A \rightarrow B} \rightarrow\text{-intro}$$

$$\frac{\Gamma, A, \dots, A \vdash \perp}{\Gamma \vdash \neg A} \neg\text{-intro}$$

$$\frac{\Gamma \vdash A \quad \Gamma \vdash B}{\Gamma \vdash A \wedge B} \wedge\text{-intro}$$

$$\frac{\Gamma \vdash A}{\Gamma \vdash A \vee B} \vee\text{-i1}$$

$$\frac{\Gamma \vdash \perp}{\Gamma \vdash A} \text{efq}$$

Implication

$$\frac{\Gamma \vdash A \rightarrow B \quad \Gamma' \vdash A}{\Gamma, \Gamma' \vdash B} \rightarrow\text{-elim}$$

Négation

$$\frac{\Gamma \vdash \neg A \quad \Gamma' \vdash A}{\Gamma, \Gamma' \vdash \perp} \neg\text{-elim}$$

Conjonction

$$\frac{\Gamma \vdash A \wedge B}{\Gamma \vdash A} \wedge\text{-elim 1} \quad \frac{\Gamma \vdash A \wedge B}{\Gamma \vdash B} \wedge\text{-elim 2}$$

Disjonction

$$\frac{\Gamma \vdash B}{\Gamma \vdash A \vee B} \vee\text{-i2} \quad \frac{\Gamma \vdash A \vee B \quad \Gamma, A, \dots, A \vdash C}{\Gamma \vdash C} \vee\text{-elim}$$

L'absurde

$$\frac{\Gamma, \neg A, \dots, \neg A \vdash \perp}{\Gamma \vdash A} \text{raa}$$

Classical Natural Deduction : version 2

Axiomes identité

$$\frac{\text{axm-id}}{\Gamma, A \vdash A, \Delta}$$

$$\frac{\Gamma, A, \dots, A \vdash B, \Delta}{\Gamma \vdash A \rightarrow B, \Delta} \rightarrow\text{-intro}$$

$$\frac{\Gamma, A, \dots, A \vdash \perp, \Delta}{\Gamma \vdash \neg A, \Delta} \neg\text{-intro}$$

$$\frac{\Gamma \vdash A, \Delta \quad \Gamma \vdash B, \Delta}{\Gamma \vdash A \wedge B, \Delta} \wedge\text{-intro}$$

$$\frac{\Gamma \vdash A, \Delta}{\Gamma \vdash A \vee B, \Delta} \vee\text{-i1}$$

$$\frac{\Gamma \vdash \perp, \Delta}{\Gamma \vdash A, \Delta} \text{efq}$$

Implication

$$\frac{\Gamma \vdash A \rightarrow B, \Delta \quad \Gamma' \vdash A, \Delta'}{\Gamma, \Gamma' \vdash B, \Delta, \Delta'} \rightarrow\text{-elim}$$

Négation

$$\frac{\Gamma \vdash \neg A, \Delta \quad \Gamma' \vdash A, \Delta}{\Gamma, \Gamma' \vdash \perp, \Delta} \neg\text{-elim}$$

Conjonction

$$\frac{\Gamma \vdash A \wedge B, \Delta}{\Gamma \vdash A, \Delta} \wedge\text{-elim 1} \quad \frac{\Gamma \vdash A \wedge B, \Delta}{\Gamma \vdash B, \Delta} \wedge\text{-elim 2}$$

Disjonction

$$\frac{\Gamma \vdash B, \Delta}{\Gamma \vdash A \vee B, \Delta} \vee\text{-i2} \quad \frac{\Gamma \vdash A \vee B, \Delta \quad \Gamma, A, \dots, A \vdash C, \Delta}{\Gamma \vdash C, \Delta}$$

L'absurde

Classical Natural Deduction : version 2

Axiomes identité

$$\frac{\text{axm-id}}{\Gamma, A \vdash A, \Delta}$$

Sequents $\Gamma \vdash \Delta$ (multi-conclusions, symmetrical)

$$\frac{\Gamma, A, \dots, A \vdash B, \Delta}{\Gamma \vdash A \rightarrow B, \Delta} \rightarrow\text{-intro}$$

Implication

$$\frac{\Gamma \vdash A \rightarrow B, \Delta \quad \Gamma' \vdash A, \Delta'}{\Gamma, \Gamma' \vdash B, \Delta, \Delta'} \rightarrow\text{-elim}$$

$$\frac{\Gamma, A, \dots, A \vdash \perp, \Delta}{\Gamma \vdash \neg A, \Delta} \neg\text{-intro}$$

Négation

$$\frac{\Gamma \vdash \neg A, \Delta \quad \Gamma' \vdash A, \Delta}{\Gamma, \Gamma' \vdash \perp, \Delta} \neg\text{-elim}$$

$$\frac{\Gamma \vdash A, \Delta \quad \Gamma \vdash B, \Delta}{\Gamma \vdash A \wedge B, \Delta} \wedge\text{-intro}$$

Conjonction

$$\frac{\Gamma \vdash A \wedge B, \Delta}{\Gamma \vdash A, \Delta} \wedge\text{-elim 1} \quad \frac{\Gamma \vdash A \wedge B, \Delta}{\Gamma \vdash B, \Delta} \wedge\text{-elim 2}$$

$$\frac{\Gamma \vdash A, \Delta}{\Gamma \vdash A \vee B, \Delta} \vee\text{-i1}$$

Disjonction

$$\frac{\Gamma \vdash B, \Delta}{\Gamma \vdash A \vee B, \Delta} \vee\text{-i2} \quad \frac{\Gamma \vdash A \vee B, \Delta \quad \Gamma, A, \dots, A \vdash C, \Delta}{\Gamma \vdash C, \Delta}$$

$$\frac{\Gamma \vdash \perp, \Delta}{\Gamma \vdash A, \Delta} \text{efq}$$

L'absurde

From Natural Deduction to Sequent Calculus

- ▶ The first dissymmetry (Hypothesis/Conclusions) disappeared :
Sequents $\Gamma \vdash \Delta$ are symmetrical

From Natural Deduction to Sequent Calculus

- ▶ The first dissymmetry (Hypothesis/Conclusions) disappeared :
Sequents $\Gamma \vdash \Delta$ are symmetrical

- ▶ Not the second dissymmetry :
Proofs continue to be “conclusion directed”

From Natural Deduction to Sequent Calculus

- ▶ The first dissymmetry (Hypothesis/Conclusions) disappeared :

Sequents $\Gamma \vdash \Delta$ are symmetrical

- ▶ Not the second dissymmetry :

Proofs continue to be “conclusion directed”

Sequent calculus : replacing elimination on the right by
Introduction on the left

Main qualities of Sequent Calculus

- ▶ Symmetries of *classical* theorems (De Morgan) now implemented in rules

Main qualities of Sequent Calculus

- ▶ Symmetries of *classical* theorems (De Morgan) now implemented in rules
- ▶ Clear architectonic of rules. They are divided in three groups :
 - ▶ Identity Group
 - ▶ Logical Group
 - ▶ Structural Group

Main qualities of Sequent Calculus

- ▶ Symmetries of *classical* theorems (De Morgan) now implemented in rules
- ▶ Clear architectonic of rules. They are divided in three groups :
 - ▶ Identity Group
 - ▶ Logical Group
 - ▶ Structural Group
- ▶ The distinction between analytic / non analytic proofs is :
 - ▶ evident
 - ▶ conceptually clear : composition of proofs

Main qualities of Sequent Calculus

- ▶ Symmetries of *classical* theorems (De Morgan) now implemented in rules
- ▶ Clear architectonic of rules. They are divided in three groups :
 - ▶ Identity Group
 - ▶ Logical Group
 - ▶ Structural Group
- ▶ The distinction between analytic / non analytic proofs is :
 - ▶ evident
 - ▶ conceptually clear : composition of proofs
- ▶ Uncover the dynamical sense of duality : Reversible / Irreversible logical rules

LK - Identity Group and Structural group

Identity group

Identity axiom

$$\text{ax} \frac{}{A \vdash A}$$

Cut rule

$$\frac{\Gamma_1 \vdash \Delta_1, A \quad A, \Gamma_2 \vdash \Delta_2}{\Gamma_1, \Gamma_2 \vdash \Delta_1, \Delta_2} \text{ cut}$$

LK - Identity Group and Structural group

Identity group

Identity axiom

$$\text{ax} \frac{}{A \vdash A}$$

Cut rule

$$\frac{\Gamma_1 \vdash \Delta_1, A \quad A, \Gamma_2 \vdash \Delta_2}{\Gamma_1, \Gamma_2 \vdash \Delta_1, \Delta_2} \text{ cut}$$

Structural group

Contractions

$$\frac{\Gamma \vdash A, A, \Delta}{\Gamma \vdash A, \Delta} \text{ ctr}$$

$$\text{ctr} \frac{\Gamma, A, A \vdash \Delta}{\Gamma, A \vdash \Delta}$$

Weakenings

$$\frac{\Gamma \vdash \Delta}{\Gamma \vdash A, \Delta} \text{ w}$$

$$\text{w} \frac{\Gamma \vdash \Delta}{\Gamma, A \vdash \Delta}$$

LK - Identity Group and Structural group

Identity group

Identity axiom

$$\text{ax} \frac{}{A \vdash A}$$

Structural group

Contractions

$$\frac{\Gamma \vdash A, A, \Delta}{\Gamma \vdash A, \Delta} \text{ctr}$$

$$\text{ctr} \frac{\Gamma, A, A \vdash \Delta}{\Gamma, A \vdash \Delta}$$

Weakenings

$$\frac{\Gamma \vdash \Delta}{\Gamma \vdash A, \Delta} \text{w}$$

$$\text{w} \frac{\Gamma \vdash \Delta}{\Gamma, A \vdash \Delta}$$

LK - Logical group

Logical group : unary connectives (negation)	
Negation	
$\frac{\Gamma, A \vdash \Delta}{\Gamma \vdash \neg A, \Delta} \neg$	$\neg \frac{\Gamma \vdash A, \Delta}{\Gamma, \neg A \vdash \Delta}$

LK - Logical group

Logical group : unary connectives (negation)

Negation

$$\frac{\Gamma, A \vdash \Delta}{\Gamma \vdash \neg A, \Delta} \neg$$

$$\neg \frac{\Gamma \vdash A, \Delta}{\Gamma, \neg A \vdash \Delta}$$

Logical group : binary dual connectives

rules for conjunction

$$\frac{\Gamma \vdash A, \Delta \quad \Gamma' \vdash B, \Delta'}{\Gamma, \Gamma' \vdash A \wedge B, \Delta, \Delta'} \wedge$$

$$\wedge \frac{\Gamma, A, B \vdash \Delta}{\Gamma, A \wedge B \vdash \Delta}$$

rules for disjunction

$$\frac{\Gamma \vdash A, B, \Delta}{\Gamma \vdash A \vee B, \Delta} \vee$$

$$\vee \frac{\Gamma, A \vdash \Delta \quad \Gamma', B \vdash \Delta'}{\Gamma, \Gamma', A \vee B \vdash \Delta, \Delta'}$$

LK - Logical group

Logical group : unary connectives (negation)

Negation

$$\frac{\Gamma, A \vdash \Delta}{\Gamma \vdash \neg A, \Delta} \neg$$

$$\neg \frac{\Gamma \vdash A, \Delta}{\Gamma, \neg A \vdash \Delta}$$

Logical group : binary dual connectives

rules for conjunction

$$\frac{\Gamma \vdash A, \Delta \quad \Gamma' \vdash B, \Delta'}{\Gamma, \Gamma' \vdash A \wedge B, \Delta, \Delta'} \wedge$$

$$\wedge \frac{\Gamma, A, B \vdash \Delta}{\Gamma, A \wedge B \vdash \Delta}$$

rules for disjunction

$$\frac{\Gamma \vdash A, B, \Delta}{\Gamma \vdash A \vee B, \Delta} \vee$$

$$\vee \frac{\Gamma, A \vdash \Delta \quad \Gamma', B \vdash \Delta'}{\Gamma, \Gamma', A \vee B \vdash \Delta, \Delta'}$$

LK - Logical group

Logical group : unary connectives (negation)

Negation

$$\frac{\Gamma, A \vdash \Delta}{\Gamma \vdash \neg A, \Delta} \neg$$

$$\neg \frac{\Gamma \vdash A, \Delta}{\Gamma, \neg A \vdash \Delta}$$

Logical group : binary dual connectives

rules for conjunction

$$\frac{\Gamma \vdash A, \Delta \quad \Gamma' \vdash B, \Delta'}{\Gamma, \Gamma' \vdash A \wedge B, \Delta, \Delta'} \wedge$$

$$\wedge \frac{\Gamma, A, B \vdash \Delta}{\Gamma, A \wedge B \vdash \Delta}$$

rules for disjunction

$$\frac{\Gamma \vdash A, B, \Delta}{\Gamma \vdash A \vee B, \Delta} \vee$$

$$\vee \frac{\Gamma, A \vdash \Delta \quad \Gamma', B \vdash \Delta'}{\Gamma, \Gamma', A \vee B \vdash \Delta, \Delta'}$$

LK - Logical group

Logical group : unary connectives (negation)

Negation

$$\frac{\Gamma, A \vdash \Delta}{\Gamma \vdash \neg A, \Delta} \neg$$

$$\neg \frac{\Gamma \vdash A, \Delta}{\Gamma, \neg A \vdash \Delta}$$

Logical group : binary dual connectives

rules for conjunction

$$\frac{\Gamma \vdash A, \Delta \quad \Gamma' \vdash B, \Delta'}{\Gamma, \Gamma' \vdash A \wedge B, \Delta, \Delta'} \wedge$$

$$\wedge \frac{\Gamma, A, B \vdash \Delta}{\Gamma, A \wedge B \vdash \Delta}$$

rules for disjunction

$$\frac{\Gamma \vdash A, B, \Delta}{\Gamma \vdash A \vee B, \Delta} \vee$$

$$\vee \frac{\Gamma, A \vdash \Delta \quad \Gamma', B \vdash \Delta'}{\Gamma, \Gamma', A \vee B \vdash \Delta, \Delta'}$$

rules for conjunction

$$\frac{\Gamma \vdash A, \Delta \quad \Gamma \vdash B, \Delta}{\Gamma \vdash A \wedge B, \Delta} \wedge$$

$$\wedge \frac{\Gamma, A \vdash \Delta}{\Gamma, A \wedge B \vdash \Delta} \wedge \frac{\Gamma, B \vdash \Delta}{\Gamma, A \wedge B \vdash \Delta}$$

rules for disjunction

$$\frac{\Gamma \vdash A, \Delta}{\Gamma \vdash A \vee B, \Delta} \vee \quad \frac{\Gamma \vdash B, \Delta}{\Gamma \vdash A \vee B, \Delta} \vee$$

$$\vee \frac{\Gamma, A \vdash \Delta \quad \Gamma, B \vdash \Delta}{\Gamma, A \vee B \vdash \Delta}$$

LK - Logical group

Logical group : unary connectives (negation)

Negation

$$\frac{\Gamma, A \vdash \Delta}{\Gamma \vdash \neg A, \Delta} \neg$$

$$\neg \frac{\Gamma \vdash A, \Delta}{\Gamma, \neg A \vdash \Delta}$$

Logical group : binary dual connectives

rules for conjunction

$$\frac{\Gamma \vdash A, \Delta \quad \Gamma' \vdash B, \Delta'}{\Gamma, \Gamma' \vdash A \wedge B, \Delta, \Delta'} \wedge$$

$$\wedge \frac{\Gamma, A, B \vdash \Delta}{\Gamma, A \wedge B \vdash \Delta}$$

rules for disjunction

$$\frac{\Gamma \vdash A, B, \Delta}{\Gamma \vdash A \vee B, \Delta} \vee$$

$$\vee \frac{\Gamma, A \vdash \Delta \quad \Gamma', B \vdash \Delta'}{\Gamma, \Gamma', A \vee B \vdash \Delta, \Delta'}$$

rules for conjunction

$$\frac{\Gamma \vdash A, \Delta \quad \Gamma \vdash B, \Delta}{\Gamma \vdash A \wedge B, \Delta} \wedge$$

$$\wedge \frac{\Gamma, A \vdash \Delta}{\Gamma, A \wedge B \vdash \Delta} \quad \wedge \frac{\Gamma, B \vdash \Delta}{\Gamma, A \wedge B \vdash \Delta}$$

rules for disjunction

$$\frac{\Gamma \vdash A, \Delta}{\Gamma \vdash A \vee B, \Delta} \vee \quad \frac{\Gamma \vdash B, \Delta}{\Gamma \vdash A \vee B, \Delta} \vee$$

$$\vee \frac{\Gamma, A \vdash \Delta \quad \Gamma, B \vdash \Delta}{\Gamma, A \vee B \vdash \Delta}$$

LK - Logical group

Logical group : unary connectives (negation)

Negation

$$\frac{\Gamma, A \vdash \Delta}{\Gamma \vdash \neg A, \Delta} \neg$$

$$\neg \frac{\Gamma \vdash A, \Delta}{\Gamma, \neg A \vdash \Delta}$$

Logical group : binary dual connectives

rules for conjunction

$$\frac{\Gamma \vdash A, \Delta \quad \Gamma' \vdash B, \Delta'}{\Gamma, \Gamma' \vdash A \wedge B, \Delta, \Delta'} \wedge$$

$$\wedge \frac{\Gamma, A, B \vdash \Delta}{\Gamma, A \wedge B \vdash \Delta}$$

rules for disjunction

$$\frac{\Gamma \vdash A, B, \Delta}{\Gamma \vdash A \vee B, \Delta} \vee$$

$$\vee \frac{\Gamma, A \vdash \Delta \quad \Gamma', B \vdash \Delta'}{\Gamma, \Gamma', A \vee B \vdash \Delta, \Delta'}$$

rules for conjunction

$$\frac{\Gamma \vdash A, \Delta \quad \Gamma \vdash B, \Delta}{\Gamma \vdash A \wedge B, \Delta} \wedge$$

$$\wedge \frac{\Gamma, A \vdash \Delta}{\Gamma, A \wedge B \vdash \Delta} \quad \wedge \frac{\Gamma, B \vdash \Delta}{\Gamma, A \wedge B \vdash \Delta}$$

rules for disjunction

$$\frac{\Gamma \vdash A, \Delta}{\Gamma \vdash A \vee B, \Delta} \vee \quad \frac{\Gamma \vdash B, \Delta}{\Gamma \vdash A \vee B, \Delta} \vee$$

$$\vee \frac{\Gamma, A \vdash \Delta \quad \Gamma, B \vdash \Delta}{\Gamma, A \vee B \vdash \Delta}$$

LK - Logical group

Logical group : unary connectives (negation)

Negation

$$\frac{\Gamma, A \vdash \Delta}{\Gamma \vdash \neg A, \Delta} \neg$$

$$\neg \frac{\Gamma \vdash A, \Delta}{\Gamma, \neg A \vdash \Delta}$$

Logical group : binary dual connectives

Multiplicative
rules for conjunction

$$\frac{\Gamma \vdash A, \Delta \quad \Gamma' \vdash B, \Delta'}{\Gamma, \Gamma' \vdash A \wedge B, \Delta, \Delta'} \wedge$$

$$\wedge \frac{\Gamma, A, B \vdash \Delta}{\Gamma, A \wedge B \vdash \Delta}$$

Multiplicative
rules for disjunction

$$\frac{\Gamma \vdash A, B, \Delta}{\Gamma \vdash A \vee B, \Delta} \vee$$

$$\vee \frac{\Gamma, A \vdash \Delta \quad \Gamma', B \vdash \Delta'}{\Gamma, \Gamma', A \vee B \vdash \Delta, \Delta'}$$

Additive
rules for conjunction

$$\frac{\Gamma \vdash A, \Delta \quad \Gamma \vdash B, \Delta}{\Gamma \vdash A \wedge B, \Delta} \wedge$$

$$\wedge \frac{\Gamma, A \vdash \Delta}{\Gamma, A \wedge B \vdash \Delta} \quad \wedge \frac{\Gamma, B \vdash \Delta}{\Gamma, A \wedge B \vdash \Delta}$$

Additive
rules for disjunction

$$\frac{\Gamma \vdash A, \Delta}{\Gamma \vdash A \vee B, \Delta} \vee \quad \frac{\Gamma \vdash B, \Delta}{\Gamma \vdash A \vee B, \Delta} \vee$$

$$\vee \frac{\Gamma, A \vdash \Delta \quad \Gamma, B \vdash \Delta}{\Gamma, A \vee B \vdash \Delta}$$

LK - Logical group

Logical group : unary connectives (negation)

Negation

$$\frac{\Gamma, A \vdash \Delta}{\Gamma \vdash \neg A, \Delta} \neg$$

$$\neg \frac{\Gamma \vdash A, \Delta}{\Gamma, \neg A \vdash \Delta}$$

Logical group : binary dual connectives

Multiplicative

rules for conjunction

$$\frac{\Gamma \vdash A, \Delta \quad \Gamma' \vdash B, \Delta'}{\Gamma, \Gamma' \vdash A \wedge B, \Delta, \Delta'} \wedge$$

$$\wedge \frac{\Gamma, A, B \vdash \Delta}{\Gamma, A \wedge B \vdash \Delta}$$

Multiplicative

rules for disjunction

$$\frac{\Gamma \vdash A, B, \Delta}{\Gamma \vdash A \vee B, \Delta} \vee$$

$$\vee \frac{\Gamma, A \vdash \Delta \quad \Gamma', B \vdash \Delta'}{\Gamma, \Gamma', A \vee B \vdash \Delta, \Delta'}$$

Additive

rules for conjunction

$$\frac{\Gamma \vdash A, \Delta \quad \Gamma \vdash B, \Delta}{\Gamma \vdash A \wedge B, \Delta} \wedge$$

$$\wedge \frac{\Gamma, A \vdash \Delta}{\Gamma, A \wedge B \vdash \Delta} \quad \wedge \frac{\Gamma, B \vdash \Delta}{\Gamma, A \wedge B \vdash \Delta}$$

Additive

rules for disjunction

$$\frac{\Gamma \vdash A, \Delta}{\Gamma \vdash A \vee B, \Delta} \vee \quad \frac{\Gamma \vdash B, \Delta}{\Gamma \vdash A \vee B, \Delta} \vee$$

$$\vee \frac{\Gamma, A \vdash \Delta \quad \Gamma, B \vdash \Delta}{\Gamma, A \vee B \vdash \Delta}$$

LK - Logical group

Logical group : unary connectives (negation)

Negation

$$\frac{\Gamma, A \vdash \Delta}{\Gamma \vdash \neg A, \Delta} \neg$$

$$\neg \frac{\Gamma \vdash A, \Delta}{\Gamma, \neg A \vdash \Delta}$$

Logical group : binary dual connectives

Multiplicative

rules for conjunction

$$\frac{\Gamma \vdash A, \Delta \quad \Gamma' \vdash B, \Delta'}{\Gamma, \Gamma' \vdash A \wedge B, \Delta, \Delta'} \wedge$$

$$\wedge \frac{\Gamma, A, B \vdash \Delta}{\Gamma, A \wedge B \vdash \Delta}$$

Multiplicative

rules for disjunction

$$\frac{\Gamma \vdash A, B, \Delta}{\Gamma \vdash A \vee B, \Delta} \vee$$

$$\vee \frac{\Gamma, A \vdash \Delta \quad \Gamma', B \vdash \Delta'}{\Gamma, \Gamma', A \vee B \vdash \Delta, \Delta'}$$

Additive

rules for conjunction

$$\frac{\Gamma \vdash A, \Delta \quad \Gamma \vdash B, \Delta}{\Gamma \vdash A \wedge B, \Delta} \wedge$$

$$\wedge \frac{\Gamma, A \vdash \Delta}{\Gamma, A \wedge B \vdash \Delta} \wedge \frac{\Gamma, B \vdash \Delta}{\Gamma, A \wedge B \vdash \Delta}$$

Additive

rules for disjunction

$$\frac{\Gamma \vdash A, \Delta}{\Gamma \vdash A \vee B, \Delta} \vee \quad \frac{\Gamma \vdash B, \Delta}{\Gamma \vdash A \vee B, \Delta} \vee$$

$$\vee \frac{\Gamma, A \vdash \Delta \quad \Gamma, B \vdash \Delta}{\Gamma, A \vee B \vdash \Delta}$$

LK - Logical group

Logical group : unary connectives (negation)

Negation

$$\frac{\Gamma, A \vdash \Delta}{\Gamma \vdash \neg A, \Delta} \neg$$

$$\neg \frac{\Gamma \vdash A, \Delta}{\Gamma, \neg A \vdash \Delta}$$

Logical group : binary dual connectives

Multiplicative
rules for conjunction

$$\frac{\Gamma \vdash A, \Delta \quad \Gamma' \vdash B, \Delta'}{\Gamma, \Gamma' \vdash A \wedge B, \Delta, \Delta'} \wedge$$

$$\wedge \frac{\Gamma, A, B \vdash \Delta}{\Gamma, A \wedge B \vdash \Delta}$$

Multiplicative
rules for disjunction

$$\frac{\Gamma \vdash A, B, \Delta}{\Gamma \vdash A \vee B, \Delta} \vee$$

$$\vee \frac{\Gamma, A \vdash \Delta \quad \Gamma', B \vdash \Delta'}{\Gamma, \Gamma', A \vee B \vdash \Delta, \Delta'}$$

Additive
rules for conjunction

$$\frac{\Gamma \vdash A, \Delta \quad \Gamma \vdash B, \Delta}{\Gamma \vdash A \wedge B, \Delta} \wedge$$

$$\wedge \frac{\Gamma, A \vdash \Delta}{\Gamma, A \wedge B \vdash \Delta} \quad \wedge \frac{\Gamma, B \vdash \Delta}{\Gamma, A \wedge B \vdash \Delta}$$

Additive
rules for disjunction

$$\frac{\Gamma \vdash A, \Delta}{\Gamma \vdash A \vee B, \Delta} \vee \quad \frac{\Gamma \vdash B, \Delta}{\Gamma \vdash A \vee B, \Delta} \vee$$

$$\vee \frac{\Gamma, A \vdash \Delta \quad \Gamma, B \vdash \Delta}{\Gamma, A \vee B \vdash \Delta}$$

LK - Logical group

Logical group : unary connectives (negation)

Negation

$$\frac{\Gamma, A \vdash \Delta}{\Gamma \vdash \neg A, \Delta} \neg$$

$$\neg \frac{\Gamma \vdash A, \Delta}{\Gamma, \neg A \vdash \Delta}$$

Logical group : binary dual connectives

Multiplicative

rules for conjunction

$$\frac{\Gamma \vdash A, \Delta \quad \Gamma' \vdash B, \Delta'}{\Gamma, \Gamma' \vdash A \overset{m}{\wedge} B, \Delta, \Delta'} \overset{m}{\wedge}$$

$$\overset{m}{\wedge} \frac{\Gamma, A, B \vdash \Delta}{\Gamma, A \overset{m}{\wedge} B \vdash \Delta}$$

Multiplicative

rules for disjunction

$$\frac{\Gamma \vdash A, B, \Delta}{\Gamma \vdash A \overset{m}{\vee} B, \Delta} \overset{m}{\vee}$$

$$\overset{m}{\vee} \frac{\Gamma, A \vdash \Delta \quad \Gamma', B \vdash \Delta'}{\Gamma, \Gamma', A \overset{m}{\vee} B \vdash \Delta, \Delta'}$$

Additive

rules for conjunction

$$\frac{\Gamma \vdash A, \Delta \quad \Gamma \vdash B, \Delta}{\Gamma \vdash A \overset{a}{\wedge} B, \Delta} \overset{a}{\wedge}$$

$$\overset{a}{\wedge} \frac{\Gamma, A \vdash \Delta}{\Gamma, A \overset{a}{\wedge} B \vdash \Delta} \quad \overset{a}{\wedge} \frac{\Gamma, B \vdash \Delta}{\Gamma, A \overset{a}{\wedge} B \vdash \Delta}$$

Additive

rules for disjunction

$$\frac{\Gamma \vdash A, \Delta}{\Gamma \vdash A \overset{a}{\vee} B, \Delta} \overset{a}{\vee} \quad \frac{\Gamma \vdash B, \Delta}{\Gamma \vdash A \overset{a}{\vee} B, \Delta} \overset{a}{\vee}$$

$$\overset{a}{\vee} \frac{\Gamma, A \vdash \Delta \quad \Gamma, B \vdash \Delta}{\Gamma, A \overset{a}{\vee} B \vdash \Delta}$$

PART 3

Classical logic decomposed

Additive and Multiplicative styles equivalent modulo the structural rules

- ▶ In presence of the structural rules, the distinction between multiplicative and additive styles degenerates, i.e. one has :

$$A \overset{a}{\vee} B \equiv A \overset{m}{\vee} B$$

$$A \overset{a}{\wedge} B \equiv A \overset{m}{\wedge} B$$

Additive and Multiplicative styles equivalent modulo the structural rules

- ▶ In presence of the structural rules, the distinction between multiplicative and additive styles degenerates, i.e. one has :

$$A \overset{a}{\vee} B \equiv A \overset{m}{\vee} B$$

$$A \overset{a}{\wedge} B \equiv A \overset{m}{\wedge} B$$

Indeed (for instance) :

$$\overset{a}{\vee} \frac{\frac{A \vdash A}{A \vdash A, B} \text{ w} \quad \frac{B \vdash B}{B \vdash A, B} \text{ w}}{A \overset{a}{\vee} B \vdash A, B} \overset{m}{\vee} \frac{A \overset{a}{\vee} B \vdash A, B}{A \overset{a}{\vee} B \vdash A \overset{m}{\vee} B} \overset{a}{\vee}$$

$$\overset{m}{\vee} \frac{A \vdash A \quad B \vdash B}{A \overset{m}{\vee} B \vdash A, B} \overset{a}{\vee} \frac{A \overset{m}{\vee} B \vdash A \overset{a}{\vee} B, B}{A \overset{m}{\vee} B \vdash A \overset{a}{\vee} B, A \overset{a}{\vee} B} \overset{a}{\vee} \frac{A \overset{m}{\vee} B \vdash A \overset{a}{\vee} B, A \overset{a}{\vee} B}{A \overset{m}{\vee} B \vdash A \overset{a}{\vee} B} \text{ ctr}$$

Additive and Multiplicative styles equivalent modulo the structural rules

- ▶ But structural rules are needed for that (easy to show once eliminability of cut is proved).

Additive and Multiplicative styles equivalent modulo the structural rules

- ▶ But structural rules are needed for that (easy to show once eliminability of cut is proved).
- ▶ So in the fragment of LK with no structural rules, **the rules with the various style define genuine (non equivalent) connectives.**

Additive and Multiplicative styles equivalent modulo the structural rules

- ▶ But structural rules are needed for that (easy to show once eliminability of cut is proved).
- ▶ So in the fragment of LK with no structural rules, **the rules with the various style define genuine (non equivalent) connectives.**
- ▶ Notation and terminology :
 - ▶ $\overset{m}{\wedge}$ noted \otimes (“tensor”, “times”)
 - ▶ $\overset{m}{\vee}$ noted \wp (“par”, “co-tensor”)
 - ▶ $\overset{a}{\wedge}$ noted $\&$ (“with”)
 - ▶ $\overset{a}{\vee}$ noted \oplus (“plus”)

MALL : Identity group

Identity group

Identity axiom

$$\text{ax} \frac{}{A \vdash A}$$

Cut rule

$$\frac{\Gamma_1 \vdash \Delta_1, A \quad A, \Gamma_2 \vdash \Delta_2}{\Gamma_1, \Gamma_2 \vdash \Delta_1, \Delta_2} \text{cut}$$

Logical group : unary connectives (negation)

Negation

$$\frac{\Gamma, A \vdash \Delta}{\Gamma \vdash \neg A, \Delta} \neg$$

$$\neg \frac{\Gamma \vdash A, \Delta}{\Gamma, \neg A \vdash \Delta}$$

Logical group : binary dual connectives

Multiplicative conjunction

(\otimes , "Tensor")

$$\frac{\Gamma \vdash A, \Delta \quad \Gamma' \vdash B, \Delta'}{\Gamma, \Gamma' \vdash A \otimes B, \Delta, \Delta'} \otimes$$

$$\otimes \frac{\Gamma, A, B \vdash \Delta}{\Gamma, A \otimes B \vdash \Delta}$$

Multiplicative disjunction

(\wp , "Par")

$$\frac{\Gamma \vdash A, B, \Delta}{\Gamma \vdash A \wp B, \Delta} \wp$$

$$\wp \frac{\Gamma, A \vdash \Delta \quad \Gamma', B \vdash \Delta'}{\Gamma, \Gamma', A \wp B \vdash \Delta, \Delta'}$$

Additive conjunction

(&, "With")

$$\frac{\Gamma \vdash A, \Delta \quad \Gamma \vdash B, \Delta}{\Gamma \vdash A \& B, \Delta} \&$$

$$\& \frac{\Gamma, A \vdash \Delta}{\Gamma, A \& B \vdash \Delta}$$

$$\& \frac{\Gamma, B \vdash \Delta}{\Gamma, A \& B \vdash \Delta}$$

Additive disjunction

(\oplus , "Plus")

$$\frac{\Gamma \vdash A, \Delta}{\Gamma \vdash A \oplus B, \Delta} \oplus \quad \frac{\Gamma \vdash B, \Delta}{\Gamma \vdash A \oplus B, \Delta} \oplus$$

$$\oplus \frac{\Gamma, A \vdash \Delta \quad \Gamma, B \vdash \Delta}{\Gamma, A \oplus B \vdash \Delta}$$

MLL : Logical group

Logical group : unary connectives (negation)

Negation

$$\frac{\Gamma, A \vdash \Delta}{\Gamma \vdash \neg A, \Delta} \neg$$

$$\neg \frac{\Gamma \vdash A, \Delta}{\Gamma, \neg A \vdash \Delta}$$

Logical group : binary dual connectives

Multiplicative conjunction

(\otimes , "Tensor")

$$\frac{\Gamma \vdash A, \Delta \quad \Gamma' \vdash B, \Delta'}{\Gamma, \Gamma' \vdash A \otimes B, \Delta, \Delta'} \otimes$$

$$\otimes \frac{\Gamma, A, B \vdash \Delta}{\Gamma, A \otimes B \vdash \Delta}$$

Multiplicative disjunction

(\wp , "Par")

$$\frac{\Gamma \vdash A, B, \Delta}{\Gamma \vdash A \wp B, \Delta} \wp$$

$$\wp \frac{\Gamma, A \vdash \Delta \quad \Gamma', B \vdash \Delta'}{\Gamma, \Gamma', A \wp B \vdash \Delta, \Delta'}$$

Additive conjunction

(&, "With")

$$\frac{\Gamma \vdash A, \Delta \quad \Gamma \vdash B, \Delta}{\Gamma \vdash A \& B, \Delta} \&$$

$$\& \frac{\Gamma, A \vdash \Delta}{\Gamma, A \& B \vdash \Delta}$$

$$\& \frac{\Gamma, B \vdash \Delta}{\Gamma, A \& B \vdash \Delta}$$

Additive disjunction

(\oplus , "Plus")

$$\frac{\Gamma \vdash A, \Delta}{\Gamma \vdash A \oplus B, \Delta} \oplus \quad \frac{\Gamma \vdash B, \Delta}{\Gamma \vdash A \oplus B, \Delta} \oplus$$

$$\oplus \frac{\Gamma, A \vdash \Delta \quad \Gamma, B \vdash \Delta}{\Gamma, A \oplus B \vdash \Delta}$$

ALL : Logical group

Logical group : unary connectives (negation)

Negation

$$\frac{\Gamma, A \vdash \Delta}{\Gamma \vdash \neg A, \Delta} \neg$$

$$\neg \frac{\Gamma \vdash A, \Delta}{\Gamma, \neg A \vdash \Delta}$$

Logical group : binary dual connectives

Multiplicative conjunction

(\otimes , "Tensor")

$$\frac{\Gamma \vdash A, \Delta \quad \Gamma' \vdash B, \Delta'}{\Gamma, \Gamma' \vdash A \otimes B, \Delta, \Delta'} \otimes$$

$$\otimes \frac{\Gamma, A, B \vdash \Delta}{\Gamma, A \otimes B \vdash \Delta}$$

Multiplicative disjunction

(\wp , "Par")

$$\frac{\Gamma \vdash A, B, \Delta}{\Gamma \vdash A \wp B, \Delta} \wp$$

$$\wp \frac{\Gamma, A \vdash \Delta \quad \Gamma', B \vdash \Delta'}{\Gamma, \Gamma', A \wp B \vdash \Delta, \Delta'}$$

Additive conjunction

(&, "With")

$$\frac{\Gamma \vdash A, \Delta \quad \Gamma \vdash B, \Delta}{\Gamma \vdash A \& B, \Delta} \&$$

$$\& \frac{\Gamma, A \vdash \Delta}{\Gamma, A \& B \vdash \Delta}$$

$$\& \frac{\Gamma, B \vdash \Delta}{\Gamma, A \& B \vdash \Delta}$$

Additive disjunction

(\oplus , "Plus")

$$\frac{\Gamma \vdash A, \Delta}{\Gamma \vdash A \oplus B, \Delta} \oplus \quad \frac{\Gamma \vdash B, \Delta}{\Gamma \vdash A \oplus B, \Delta} \oplus$$

$$\oplus \frac{\Gamma, A \vdash \Delta \quad \Gamma, B \vdash \Delta}{\Gamma, A \oplus B \vdash \Delta}$$

MALL : adding 0-ary connectives (neutrals)

The language is enriched with four 0-ary connectives (thus formulas) :

Multiplicative ones : 1 and \perp Additive ones : 0 and \top

MALL : adding 0-ary connectives (neutrals)

The language is enriched with four 0-ary connectives (thus formulas) :

Multiplicative ones : 1 and \perp

Additive ones : 0 and \top

MALL Logical group continued: Neutrals	
0-ary multiplicatives (neutrals)	
$\frac{}{\vdash 1} 1$	$\perp \frac{\Gamma \vdash \Delta}{\Gamma, 1 \vdash \Delta}$
-----	-----
$\frac{\Gamma \vdash \Delta}{\Gamma \vdash \Delta, \perp} \perp$	$\perp \frac{}{\perp \vdash}$
0-ary additives (neutrals)	
$\frac{}{\Gamma \vdash \Delta, \top} \top$	<i>No left intro for \top</i>
-----	-----
<i>No right rule for 0</i>	$0 \frac{}{\Gamma, 0 \vdash \Delta}$

MALL : adding 0-ary connectives (neutrals)

The language is enriched with four 0-ary connectives (thus formulas) :

Multiplicative ones : 1 and \perp

Additive ones : 0 and \top

MALL Logical group continued: Neutrals	
0-ary multiplicatives (neutrals)	
$\frac{}{\vdash 1} 1$	$\perp \frac{\Gamma \vdash \Delta}{\Gamma, 1 \vdash \Delta}$
-----	-----
$\frac{\Gamma \vdash \Delta}{\Gamma \vdash \Delta, \perp} \perp$	$\perp \frac{}{\perp \vdash}$
0-ary additives (neutrals)	
$\frac{}{\Gamma \vdash \Delta, \top} \top$	<i>No left intro for \top</i>
-----	-----
<i>No right rule for 0</i>	$0 \frac{}{\Gamma, 0 \vdash \Delta}$

They are neutrals :

▶ 1 is provably neutral for \otimes

\perp is provably neutral for \wp

▶ \top is provably neutral for $\&$

0 is provably neutral for \oplus

MLL, ALL, MALL are computational fragments

Each of MLL, ALL (and thus MALL) satisfies :

- ▶ **Cut-elimination** (no contractions \Rightarrow low complexity process)

MLL, ALL, MALL are computational fragments

Each of MLL, ALL (and thus MALL) satisfies :

- ▶ **Cut-elimination** (no contractions \Rightarrow low complexity process)
- ▶ **Atomization of axioms**, i.e. Identities are canonically provable from atomic initial sequents :

$$\otimes \frac{\frac{A \vdash A \quad B \vdash B}{A, B \vdash A \otimes B} \otimes}{A \otimes B \vdash A \otimes B} \otimes \qquad \wp \frac{\frac{A \vdash A \quad B \vdash B}{A \wp B \vdash A, B} \wp}{A \wp B \vdash A \wp B} \wp$$

$$\& \frac{\frac{A \vdash A}{A \& B \vdash A} \& \quad \frac{B \vdash B}{A \& B \vdash B} \&}{A \& B \vdash A \& B} \& \qquad \oplus \frac{\frac{A \vdash A}{A \vdash A \oplus B} \oplus \quad \frac{B \vdash B}{B \vdash A \oplus B} \oplus}{A \oplus B \vdash A \oplus B} \oplus$$

De Morgan dualities

- ▶ Each fragment MLL, ALL (and thus MALL) satisfies De Morgan equivalences:

$$\neg(A \otimes B) \equiv_{\text{MLL}} \neg A \wp \neg B \qquad \neg(A \wp B) \equiv_{\text{MLL}} \neg A \otimes \neg B$$

$$\neg 1 \equiv_{\text{MLL}} \perp \qquad \neg \perp \equiv_{\text{MLL}} 1$$

$$\neg(A \& B) \equiv_{\text{ALL}} \neg A \oplus \neg B \qquad \neg(A \oplus B) \equiv_{\text{ALL}} \neg A \& \neg B$$

$$\neg \top \equiv_{\text{ALL}} 0 \qquad \neg 0 \equiv_{\text{ALL}} \top$$

De Morgan dualities

- ▶ Each fragment MLL, ALL (and thus MALL) satisfies De Morgan equivalences:

$$\neg(A \otimes B) \equiv_{\text{MLL}} \neg A \wp \neg B \qquad \neg(A \wp B) \equiv_{\text{MLL}} \neg A \otimes \neg B$$

$$\neg 1 \equiv_{\text{MLL}} \perp \qquad \neg \perp \equiv_{\text{MLL}} 1$$

$$\neg(A \& B) \equiv_{\text{ALL}} \neg A \oplus \neg B \qquad \neg(A \oplus B) \equiv_{\text{ALL}} \neg A \& \neg B$$

$$\neg \top \equiv_{\text{ALL}} 0 \qquad \neg 0 \equiv_{\text{ALL}} \top$$

- ▶ Pairs of mutually dual connectives :

$$\otimes / \wp \quad \oplus / \& \quad 1 / \perp \quad 0 / \top \quad (\text{ and } \forall / \exists)$$

De Morgan dualities

- ▶ Each fragment MLL, ALL (and thus MALL) satisfies De Morgan equivalences:

$$\neg(A \otimes B) \equiv_{\text{MLL}} \neg A \wp \neg B \qquad \neg(A \wp B) \equiv_{\text{MLL}} \neg A \otimes \neg B$$

$$\neg 1 \equiv_{\text{MLL}} \perp \qquad \neg \perp \equiv_{\text{MLL}} 1$$

$$\neg(A \& B) \equiv_{\text{ALL}} \neg A \oplus \neg B \qquad \neg(A \oplus B) \equiv_{\text{ALL}} \neg A \& \neg B$$

$$\neg \top \equiv_{\text{ALL}} 0 \qquad \neg 0 \equiv_{\text{ALL}} \top$$

- ▶ Pairs of mutually dual connectives :

$$\otimes / \wp \quad \oplus / \& \quad 1 / \perp \quad 0 / \top \quad (\text{ and } \forall / \exists)$$

- ▶ Symmetry or chattering ? Up to the exchanges left/right and the exchange of dual connectives, everything is said twice. For instance :

Ceasing chattering: toward monolateral sequent calculus

Goal: to divide the number of rules for binary connectives by two.

Tool: replace negation as a unary function by the binary relation of duality.

Ceasing chattering: toward monolateral sequent calculus

Goal: to divide the number of rules for binary connectives by two.

Tool: replace negation as a unary function by the binary relation of duality.

- ▶ **Step 1:** we change the notion of formulas (our old set of formulas will be quotiented by de Morgan equivalences):
 - ▶ Negation no more present as a connective, but as a defined operation $(.)^\perp$
 - ▶ Atoms come by pairs: each atom X comes with its dual X^\perp
 - ▶ $(X^\perp)^\perp = X$
 - ▶ The dual A^\perp of A is “the De Morganized” form of $\neg A$
 - ▶ For instance : if $A = (X \otimes (\perp Y \& Z))$, then A^\perp denotes $(X^\perp \wp (Y \oplus Z^\perp))$

Ceasing chattering: toward monolateral sequent calculus

Goal: to divide the number of rules for binary connectives by two.

Tool: replace negation as a unary function by the binary relation of duality.

- ▶ **Step 1:** we change the notion of formulas (our old set of formulas will be quotiented by de Morgan equivalences):
 - ▶ Negation no more present as a connective, but as a defined operation $(.)^\perp$
 - ▶ Atoms come by pairs: each atom X comes with its dual X^\perp
 - ▶ $(X^\perp)^\perp = X$
 - ▶ The dual A^\perp of A is “the De Morganized” form of $\neg A$
 - ▶ For instance : if $A = (X \otimes (\perp Y \& Z))$, then A^\perp denotes $(X^\perp \wp (Y \oplus Z^\perp))$
- ▶ **Step 2 :** we fold the left side on the right side up; and, in the “Identity group”, we replace **identity constraints** on formulas (the one on the hypothesis side, the one on the conclusions side) by **duality constraints** (on the conclusions side)

MALL, MLL, ALL as monolateral systems

Identity group (better called : duality group)

$$\frac{}{\vdash A, A^\perp} \text{ ax}$$

$$\frac{\vdash \Gamma, A \quad \vdash \Delta, A^\perp}{\vdash \Gamma, \Delta} \text{ cut}$$

Logical group (dual connectives)

Multiplicatives

Binary multiplicatives

$$\frac{\vdash A, \Gamma \quad \vdash B, \Delta}{\vdash A \otimes B, \Gamma, \Delta} \otimes$$

$$\frac{\vdash A, B, \Gamma}{\vdash A \wp B, \Gamma} \wp$$

0-ary multiplicatives (neutrals)

$$\frac{}{\vdash 1} 1$$

$$\frac{\vdash \Gamma}{\vdash \Gamma, \perp} \perp$$

Additives

$$\frac{\vdash A, \Gamma \quad \vdash B, \Gamma}{\vdash A \& B, \Gamma} \&$$

$$\frac{\vdash A, \Gamma}{\vdash A \oplus B, \Gamma} \oplus \quad \frac{\vdash B, \Gamma}{\vdash A \oplus B, \Gamma} \oplus$$

0-ary additives (neutrals)

$$\frac{}{\vdash \top, \Gamma} \top$$

No rule for 0

Toward a reintroduction of structural rules

- ▶ Multiplicatives and Additive now coexist while being separated.

Toward a reintroduction of structural rules

- ▶ Multiplicatives and Additive now coexist while being separated.
- ▶ A price is paid : forgetting the structural rules = losing algorithmic expressivity

Toward a reintroduction of structural rules

- ▶ Multiplicatives and Additive now coexist while being separated.
- ▶ A price is paid : forgetting the structural rules = losing algorithmic expressivity
- ▶ Indeed, dynamically (i.e. w.r.t. the cut-elimination process):
 - ▶ contraction rule = duplication (the source of the complexity of cut-elimination)
 - ▶ weakening rule = erasing

Toward a reintroduction of structural rules

- ▶ Multiplicatives and Additive now coexist while being separated.
- ▶ A price is paid : forgetting the structural rules = losing algorithmic expressivity
- ▶ Indeed, dynamically (i.e. w.r.t. the cut-elimination process):
 - ▶ contraction rule = duplication (the source of the complexity of cut-elimination)
 - ▶ weakening rule = erasing
- ▶ In order to recover algorithmical richness and complexity : to reintroduce somehow structural rules is required

Toward a reintroduction of structural rules

- ▶ Multiplicatives and Additive now coexist while being separated.
- ▶ A price is paid : forgetting the structural rules = losing algorithmic expressivity
- ▶ Indeed, dynamically (i.e. w.r.t. the cut-elimination process):
 - ▶ contraction rule = duplication (the source of the complexity of cut-elimination)
 - ▶ weakening rule = erasing
- ▶ In order to recover algorithmical richness and complexity : to reintroduce somehow structural rules is required
- ▶ Problem : how to reintroduce them avoiding the collapse of (multiplicative/additive) styles ?

Toward a reintroduction of structural rules

- ▶ Multiplicatives and Additive now coexist while being separated.
- ▶ A price is paid : forgetting the structural rules = loosing algorithmic expressivity
- ▶ Indeed, dynamically (i.e. w.r.t. the cut-elimination process):
 - ▶ contraction rule = duplication (the source of the complexity of cut-elimination)
 - ▶ weakening rule = erasing
- ▶ In order to recover algorithmical richness and complexity : to reintroduce somehow structural rules is required
- ▶ Problem : how to reintroduce them avoiding the collapse of (multiplicative/additive) styles ?
- ▶ LL answer :
 - ▶ introduce “aspect” in the logical language
 - ▶ Aspect : a category coming from natural languages grammar

Exponentials

- ▶ Imperfect = durable (reusable). Perfect = Instantaneous (non reusable)

Exponentials

- ▶ Imperfect = durable (reusable). Perfect = Instantaneous (non reusable)
- ▶ Reminder : the LK contraction rule :

$$\frac{\vdash \Gamma, A, A}{\vdash \Gamma, A} \text{ctr}$$

Exponentials

- ▶ Imperfect = durable (reusable). Perfect = Instantaneous (non reusable)

- ▶ Reminder : the LK contraction rule :

$$\frac{\vdash \Gamma, A, A}{\vdash \Gamma, A} \text{ctr}$$

- ▶ From a proof search point of view (i.e. a Bottom-up reading), that rule says : one can “save” A . In other words : any A is reusable.

Exponentials

- ▶ Imperfect = durable (reusable). Perfect = Instantaneous (non reusable)

- ▶ Reminder : the LK contraction rule :

$$\frac{\vdash \Gamma, A, A}{\vdash \Gamma, A} \text{ctr}$$

- ▶ From a proof search point of view (i.e. a Bottom-up reading), that rule says : one can “save” A . In other words : any A is reusable.
- ▶ In LL, a formula is “reusable” **only** if it is marked with the seal (noted : ?) of imperfection

Exponentials

- ▶ Imperfect = durable (reusable). Perfect = Instantaneous (non reusable)
- ▶ Reminder : the LK contraction rule :

$$\frac{\vdash \Gamma, A, A}{\vdash \Gamma, A} \text{ctr}$$

- ▶ From a proof search point of view (i.e. a Bottom-up reading), that rule says : one can “save” A . In other words : any A is reusable.
- ▶ In LL, a formula is “reusable” **only** if it is marked with the seal (noted : ?) of imperfection
- ▶ Structural rules of LL :

$$\frac{\vdash \Gamma, ?A, ?A}{\vdash \Gamma, ?A} \text{ctr}$$

$$\frac{\vdash \Gamma}{\vdash \Gamma, ?A} \text{w}$$

Exponentials

- ▶ Imperfect = durable (reusable). Perfect = Instantaneous (non reusable)
- ▶ Reminder : the LK contraction rule :

$$\frac{\vdash \Gamma, A, A}{\vdash \Gamma, A} \text{ctr}$$

- ▶ From a proof search point of view (i.e. a Bottom-up reading), that rule says : one can “save” A . In other words : any A is reusable.
- ▶ In LL, a formula is “reusable” **only** if it is marked with the seal (noted : ?) of imperfection
- ▶ Structural rules of LL :

$$\frac{\vdash \Gamma, ?A, ?A}{\vdash \Gamma, ?A} \text{ctr}$$

$$\frac{\vdash \Gamma}{\vdash \Gamma, ?A} \text{w}$$

- ▶ The dual of ? will be noted ! (they are unary connectives or “modalities”)

LL (as a monolateral sequent calculus)

IDENTITY GROUP or better: **DUALITY GROUP**

$$\frac{}{\vdash A, \perp A} \text{ ax}$$

$$\frac{\vdash \Gamma, A \quad \vdash \Delta, \perp A}{\vdash \Gamma, \Delta} \text{ cut}$$

LOGICAL GROUP

Multiplicatives

2-ary multiplicatives

$$\frac{\vdash A, \Gamma \quad \vdash B, \Delta}{\vdash A \otimes B, \Gamma, \Delta} \otimes$$

$$\frac{\vdash A, B, \Gamma}{\vdash A \wp B, \Gamma} \wp$$

0-ary multiplicatives (neutrals)

$$\frac{}{\vdash 1} 1$$

$$\frac{\vdash \Gamma}{\vdash \Gamma, \perp} \perp$$

Additives

2-ary additives

$$\frac{\vdash A, \Gamma \quad \vdash B, \Gamma}{\vdash A \& B, \Gamma} \&$$

$$\frac{\vdash A, \Gamma}{\vdash A \oplus B, \Gamma} \oplus$$

$$\frac{\vdash B, \Gamma}{\vdash A \oplus B, \Gamma} \oplus$$

0-ary additives (neutrals)

$$\frac{}{\vdash \Gamma, \top} \top$$

No rule for 0

Exponentials

$$\frac{\vdash A, \Gamma}{\vdash ?A, \Gamma} ?$$

$$\frac{\vdash ?\Gamma, A}{\vdash ?\Gamma, !A} !$$

STRUCTURAL GROUP

$$\frac{\vdash \Gamma, ?A, ?A}{\vdash \Gamma, ?A} \text{ ctr}$$

$$\frac{\vdash \Gamma}{\vdash \Gamma, ?A} \text{ w}$$