Lecture 3:
From linearity in coherence spaces to Linear Logic

Lionel Vaux

Institut de Mathématiques de Luminy, Université d'Aix-Marseille, France

School on Proof Theory
Paraty 2012

A computational point of view on coherence semantics

An alternative presentation:

$$\frac{}{\Delta^{\emptyset}, x^{\{\alpha\}} : A \vdash x^{\alpha} : A} \qquad \frac{\Gamma, x^{a} : A \vdash s^{\beta} : B}{\Gamma \vdash \lambda x\, s^{(a,\beta)} : A \to B}$$

$$\frac{\Gamma_0 \vdash s^{(\{\alpha_1,\dots,\alpha_k\},\beta)} : A \to B \qquad \Gamma_1 \vdash t^{\alpha_1} : A \quad \cdots \quad \Gamma_k \vdash t^{\alpha_k} : A \quad (*)}{\bigcup_{j=0}^{k} \Gamma_j \vdash (s)\, t^{\beta} : B}$$

$$\frac{\Gamma \vdash {s_1}^{\alpha} : A_1 \qquad \Gamma \vdash s_2 : A_2}{\Gamma \vdash \langle s_1, s_2\rangle^{(1,\alpha)} : A_1 \times A_2} \quad \frac{\Gamma \vdash s_1 : A_1 \qquad \Gamma \vdash {s_2}^{\alpha} : A_2}{\Gamma \vdash \langle s_1, s_2\rangle^{(2,\alpha)} : A_1 \times A_2}$$

$$\frac{\Gamma \vdash s^{(i,\alpha)} : A_1 \times A_2}{\Gamma \vdash \pi_i s^{\alpha} : A_i}$$

A computational point of view on coherence semantics

An alternative presentation:

$$\frac{}{\Delta^{\emptyset}, x^{\{\alpha\}} : A \vdash x^{\alpha} : A} \qquad \frac{\Gamma, x^{a} : A \vdash s^{\beta} : B}{\Gamma \vdash \lambda x\, s^{(a,\beta)} : A \to B}$$

$$\frac{\Gamma_0 \vdash s^{(\{\alpha_1,\ldots,\alpha_k\},\beta)} : A \to B \qquad \Gamma_1 \vdash t^{\alpha_1} : A \quad \cdots \quad \Gamma_k \vdash t^{\alpha_k} : A}{\bigcup_{j=0}^{k} \Gamma_j \vdash (s)\, t^{\beta} : B} \ (*)$$

$$\frac{\Gamma \vdash {s_1}^{\alpha} : A_1 \qquad \Gamma \vdash s_2 : A_2}{\Gamma \vdash \langle s_1, s_2\rangle^{(1,\alpha)} : A_1 \times A_2} \quad \frac{\Gamma \vdash s_1 : A_1 \qquad \Gamma \vdash {s_2}^{\alpha} : A_2}{\Gamma \vdash \langle s_1, s_2\rangle^{(2,\alpha)} : A_1 \times A_2}$$

$$\frac{\Gamma \vdash s^{(i,\alpha)} : A_1 \times A_2}{\Gamma \vdash \pi_i s^{\alpha} : A_i}$$

$(*)$ Coherence: $\{\alpha_1, \ldots, \alpha_k\}$ and the labels of $\bigcup_{j=0}^{k} \Gamma_j$ must be cliques.

The stable function $[\![x_1 : A_1, \ldots, x_n : A_n \vdash s : B]\!]$ has trace:

$$\left\{(a_1 + \ldots + a_n, \beta);\ x_1^{a_1} : A_1, \ldots, x_n^{a_n} : A_n \vdash s^{\beta} : B\right\}$$

Linearity: algebraically

Definition

A stable function $f : \mathcal{A} \to \mathcal{B}$ is linear if it preserves coherent unions:

$$a \cup a' \in \mathcal{A} \text{ implies } f\left(a \cup a'\right) = f(a) \cup f(a')$$

and

$$f(\emptyset) = \emptyset$$

Linearity: algebraically

Definition

A stable function $f : \mathcal{A} \to \mathcal{B}$ is linear if it preserves coherent unions:

$$a \cup a' \in \mathcal{A} \text{ implies } f\left(a \cup a'\right) = f(a) \cup f(a')$$

and

$$f(\emptyset) = \emptyset$$

Consider $\cup$ (resp. $\emptyset$) as a "qualitative" counterpart to $+$ (resp. 0).

Linearity: computationally

Definition

A stable function f is linear iff all the elements of $\mathcal{T}r(f)$ are of the form $(\{\alpha\}, \beta)$.

$\rightsquigarrow$ Show the equivalence with the previous definition.

Linearity: computationally

Definition

A stable function f is linear iff all the elements of $\mathcal{T}r(f)$ are of the form $(\{\alpha\}, \beta)$.

$\rightsquigarrow$ Show the equivalence with the previous definition.

Computational linearity: f uses its argument exactly once.

Linearity: computationally

Definition

A stable function f is linear iff all the elements of $\mathcal{T}r(f)$ are of the form $(\{\alpha\}, \beta)$.

$\rightsquigarrow$ Show the equivalence with the previous definition.

Computational linearity: f uses its argument exactly once.

Example

Consider the expanded identity
$\lambda x\, \lambda y\, (x)\, y : (A \to B) \to A \to B$: it is linear in x

$\rightsquigarrow$ Compute the trace.

Linearity: computationally

Definition

A stable function f is linear iff all the elements of $\mathcal{T}r(f)$ are of the form $(\{\alpha\}, \beta)$.

$\rightsquigarrow$ Show the equivalence with the previous definition.

Computational linearity: f "uses its argument exactly once."

Example

Consider the expanded identity
$\lambda x\, \lambda y\, (x)\, y : (A \to B) \to A \to B$: it is linear in x but not necessarily in y (although y has exactly one occurrence).

$\rightsquigarrow$ Compute the trace.

Examples

The identity function is linear

No big deal.

Examples

The identity function is linear

No big deal.

Application is linear in the function

Fix some $a \in \mathcal{A}$. The (stable) operator $f \mapsto f(a)$ from $\mathcal{A} \to \mathcal{B}$ to $\mathcal{B}$ is linear.

Examples

The identity function is linear

No big deal.

Application is linear in the function

Fix some $a \in \mathcal{A}$. The (stable) operator $f \mapsto f(a)$ from $\mathcal{A} \to \mathcal{B}$ to $\mathcal{B}$ is linear.

"Applicators"

All terms of the form $\lambda x\, s$ such that x occurs only as the head variable of s, have a linear semantics.

$\rightsquigarrow$ Easy from the rules.

Projections

The projections $\mathcal{A}_1 \mathbin{\&} \mathcal{A}_2 \to \mathcal{A}_i$ are linear.

Linear trace

Definition

Let f be a linear function from $\mathcal{A}$ to $\mathcal{B}$. The linear trace of f is:

$$\mathcal{T}r_l(f) = \{(\alpha, \beta); \beta \in f(\{\alpha\})\}$$

In other words:

$$\mathcal{T}r_l(f) = \{(\alpha, \beta); (\{\alpha\}, \beta) \in \mathcal{T}r(f)\}\,.$$

Clearly, if f is linear, then $f(a) = \mathcal{T}r_l(f) \cdot a$,
where $\cdot$ denotes the straightforward relation composition.

Examples

$\mathcal{T}r_l(\lambda x\, x) = \{(\alpha, \alpha); \alpha \in |\mathcal{A}|\}$.

$\leadsto \mathcal{T}r_l(\lambda x\, \lambda y\, (x)\, y\, y) = ?$

Another key example: (graphs of) rigid embeddings are linear traces.

$$\multimap$$

linear implication, lollypop

Linear implication

Recall that...

Traces of stable functions $\mathcal{A} \to \mathcal{B}$ form a coherence space with web $\mathcal{A}_{\text{fin}} \times |\mathcal{B}|$ and such that $(a, \beta) \mathrel{\overset{\frown}{\smile}}_{\mathcal{A}\to\mathcal{B}} (a', \beta')$ iff:

- $a \cup a' \in \mathcal{A}$ implies $\beta \mathrel{\overset{\frown}{\smile}}_{\mathcal{B}} \beta'$
- if moreover $a \neq a'$, then $\beta \frown_{\mathcal{B}} \beta'$.

Linear implication

Recall that...

Traces of stable functions $\mathcal{A} \to \mathcal{B}$ form a coherence space with web $\mathcal{A}_{\text{fin}} \times |\mathcal{B}|$ and such that $(a, \beta) \frown\!\!\!\!\smile_{\mathcal{A}\to\mathcal{B}} (a', \beta')$ iff:

- $a \cup a' \in \mathcal{A}$ implies $\beta \frown\!\!\!\!\smile_{\mathcal{B}} \beta'$
- if moreover $a \neq a'$, then $\beta \frown_{\mathcal{B}} \beta'$.

Linear functions actually form a coherence space by themselves:

Definition

Let $\mathcal{A} \multimap \mathcal{B}$ be the coherence space with web $|\mathcal{A}| \times |\mathcal{B}|$, and such that: $(\alpha, \beta) \frown\!\!\!\!\smile_{\mathcal{A}\multimap\mathcal{B}} (\alpha', \beta')$ iff:

- $\alpha \frown\!\!\!\!\smile_{\mathcal{A}} \alpha'$ implies $\beta \frown\!\!\!\!\smile_{\mathcal{B}} \beta'$
- if moreover $a \neq a'$, then $\beta \frown_{\mathcal{B}} \beta'$.

Linear implication

Recall that...

Traces of stable functions $\mathcal{A} \to \mathcal{B}$ form a coherence space with web $\mathcal{A}_{\text{fin}} \times |\mathcal{B}|$ and such that $(a, \beta) \frown\!\!\!\!\smile_{\mathcal{A}\to\mathcal{B}} (a', \beta')$ iff:

- $a \cup a' \in \mathcal{A}$ implies $\beta \frown\!\!\!\!\smile_{\mathcal{B}} \beta'$
- if moreover $a \neq a'$, then $\beta \frown_{\mathcal{B}} \beta'$.

Linear functions actually form a coherence space by themselves:

Definition

Let $\mathcal{A} \multimap \mathcal{B}$ be the coherence space with web $|\mathcal{A}| \times |\mathcal{B}|$, and such that: $(\alpha, \beta) \frown\!\!\!\!\smile_{\mathcal{A}\multimap\mathcal{B}} (\alpha', \beta')$ iff:

- $\alpha \frown\!\!\!\!\smile_{\mathcal{A}} \alpha'$ implies $\beta \frown\!\!\!\!\smile_{\mathcal{B}} \beta'$
- if moreover $a \neq a'$, then $\beta \frown_{\mathcal{B}} \beta'$.

Linear implication

Recall that…

Traces of stable functions $\mathcal{A} \to \mathcal{B}$ form a coherence space with web $\mathcal{A}_{\text{fin}} \times |\mathcal{B}|$ and such that $(a, \beta) \coh_{\mathcal{A}\to\mathcal{B}} (a', \beta')$ iff:

- $a \cup a' \in \mathcal{A}$ implies $\beta \coh_{\mathcal{B}} \beta'$
- if moreover $a \neq a'$, then $\beta \frown_{\mathcal{B}} \beta'$.

Linear functions actually form a coherence space by themselves:

Definition

Let $\mathcal{A} \multimap \mathcal{B}$ be the coherence space with web $|\mathcal{A}| \times |\mathcal{B}|$, and such that: $(\alpha, \beta) \coh_{\mathcal{A}\multimap\mathcal{B}} (\alpha', \beta')$ iff:

- $\alpha \coh_{\mathcal{A}} \alpha'$ implies $\beta \coh_{\mathcal{B}} \beta'$
- $\alpha \frown_{\mathcal{A}} \alpha'$ implies $\beta \frown_{\mathcal{B}} \beta'$.

Linear implication

Recall that...

Traces of stable functions $\mathcal{A} \to \mathcal{B}$ form a coherence space with web $\mathcal{A}_{\mathrm{fin}} \times |\mathcal{B}|$ and such that $(a, \beta) \frown_{\mathcal{A}\to\mathcal{B}} (a', \beta')$ iff:

- $a \cup a' \in \mathcal{A}$ implies $\beta \frown_{\mathcal{B}} \beta'$
- if moreover $a \neq a'$, then $\beta \frown_{\mathcal{B}} \beta'$.

Linear functions actually form a coherence space by themselves:

Definition

Let $\mathcal{A} \multimap \mathcal{B}$ be the coherence space with web $|\mathcal{A}| \times |\mathcal{B}|$, and such that: $(\alpha, \beta) \frown_{\mathcal{A}\multimap\mathcal{B}} (\alpha', \beta')$ iff:

- $\alpha \frown_{\mathcal{A}} \alpha'$ implies $\beta \frown_{\mathcal{B}} \beta'$
- $\beta \asymp_{\mathcal{B}} \beta'$ implies $\alpha \asymp_{\mathcal{B}} \alpha'$.

!

"of course", bang

Linearization of stable maps

The similarity between the conditions for $\multimap$:

- $\alpha \frown\!\!\!\!\smile_{\mathcal{A}} \alpha'$ implies $\beta \frown\!\!\!\!\smile_{\mathcal{B}} \beta'$
- $\alpha \frown_{\mathcal{A}} \alpha'$ implies $\beta \frown_{\mathcal{B}} \beta'$.

and those for $\rightarrow$:

- $a \cup a' \in \mathcal{A}$ implies $\beta \frown\!\!\!\!\smile_{\mathcal{B}} \beta'$
- if moreover $a \neq a'$, then $\beta \frown_{\mathcal{B}} \beta'$.

suggests the introduction of a space of finite cliques $!\mathcal{A}$:

Definition

The space $!\mathcal{A}$ has web $\mathcal{A}_{\mathrm{fin}}$ and coherence:

$$a \frown\!\!\!\!\smile_{!\mathcal{A}} a' \iff a \cup a' \in \mathcal{A}.$$

Linearization of stable maps

The similarity between the conditions for $\multimap$:

- $\alpha \frown\!\!\!\!\smile_{\mathcal{A}} \alpha'$ implies $\beta \frown\!\!\!\!\smile_{\mathcal{B}} \beta'$
- $\alpha \frown_{\mathcal{A}} \alpha'$ implies $\beta \frown_{\mathcal{B}} \beta'$.

and those for $\to$:

- $a \cup a' \in \mathcal{A}$ implies $\beta \frown\!\!\!\!\smile_{\mathcal{B}} \beta'$
- if moreover $a \neq a'$, then $\beta \frown_{\mathcal{B}} \beta'$.

suggests the introduction of a space of finite cliques $!\mathcal{A}$:

Definition

The space $!\mathcal{A}$ has web $\mathcal{A}_{\text{fin}}$ and coherence:

$$a \frown\!\!\!\!\smile_{!\mathcal{A}} a' \iff a \cup a' \in \mathcal{A}.$$

Then $\mathcal{A} \to \mathcal{B} = !\mathcal{A} \multimap \mathcal{B}$, and stable functions from $\mathcal{A} \to \mathcal{B}$ "are" linear functions from $!\mathcal{A}$ to $\mathcal{B}$.

$\rightsquigarrow$ Describe the bijection extensionally.

$$\left(\cdot \right)^{\perp}$$

linear negation, dual, polar, orthogonal

Linear negation

The symmetry of conditions:

- $\alpha \frown_{\mathcal{A}} \alpha'$ implies $\beta \frown_{\mathcal{B}} \beta'$
- $\beta \asymp_{\mathcal{B}} \beta'$ implies $\alpha \asymp_{\mathcal{B}} \alpha'$

suggests the introduction of linear negation:

Definition

The dual space of $\mathcal{A}$, denoted $\mathcal{A}^{\perp}$, has web $|\mathcal{A}|$ and coherence:

$$\alpha \frown_{\mathcal{A}^{\perp}} \alpha' \iff \alpha \asymp_{\mathcal{A}} \alpha'$$

Transposition

Clearly $\mathcal{A}^{\perp\perp} = \mathcal{A}$ and there is a linear involution from $\mathcal{A} \multimap \mathcal{B}$ to $\mathcal{B}^{\perp} \multimap \mathcal{A}^{\perp}$.

(we could even write $\mathcal{A} \multimap \mathcal{B} = \mathcal{A}^{\perp} \multimapinv \mathcal{B}^{\perp}$)

$\leadsto$ Find the (linear) trace of this involution...

Linear negation

The symmetry of conditions:

- $\alpha \frown_{\mathcal{A}} \alpha'$ implies $\beta \frown_{\mathcal{B}} \beta'$
- $\beta \asymp_{\mathcal{B}} \beta'$ implies $\alpha \asymp_{\mathcal{B}} \alpha'$

suggests the introduction of linear negation:

Definition

The dual space of $\mathcal{A}$, denoted $\mathcal{A}^{\perp}$, has web $|\mathcal{A}|$ and coherence:

$$\alpha \frown_{\mathcal{A}^{\perp}} \alpha' \iff \alpha \asymp_{\mathcal{A}} \alpha'$$

Transposition

Clearly $\mathcal{A}^{\perp\perp} = \mathcal{A}$ and there is a linear involution from $\mathcal{A} \multimap \mathcal{B}$ to $\mathcal{B}^{\perp} \multimap \mathcal{A}^{\perp}$.

(we could even write $\mathcal{A} \multimap \mathcal{B} = \mathcal{A}^{\perp} \multimapinv \mathcal{B}^{\perp}$)

$\leadsto$ Find the (linear) trace of this involution...
(hint in the paragraph title)

Linear logic?

The story so far

- logic: λ-calculus, system F, ...

Linear logic?

The story so far

- logic: λ-calculus, system F, ...
- semantics: stable functions between coherence spaces

Linear logic?

The story so far

- logic: λ-calculus, system F, ...
- semantics: stable functions between coherence spaces
- decomposition of implication semantically

Linear logic?

The story so far

- logic: λ-calculus, system F, ...
- semantics: stable functions between coherence spaces
- decomposition of implication semantically

Is there a logical system behind that refinement, based on linear implication?

Linear category of coherence spaces

Linear functions compose

More precisely, coherence spaces and linear functions between them form a category $\mathbf{Coh}_l$.

Product

The direct product & is still a categorical product (pairing)

Linear category of coherence spaces

Linear functions compose

More precisely, coherence spaces and linear functions between them form a category $\mathbf{Coh}_l$.

Product

The direct product & is still a categorical product (pairing) but not properly related with $\multimap$:
$\mathcal{A} \,\&\, \mathcal{B} \multimap \mathcal{C}$ is not the type of bilinear functions.

Linear category of coherence spaces

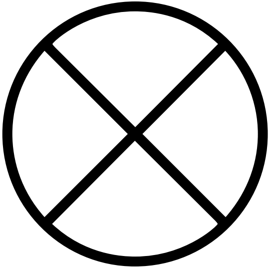
Linear functions compose

More precisely, coherence spaces and linear functions between them form a category $\mathbf{Coh}_l$.

Product

The direct product & is still a categorical product (pairing) but not properly related with $\multimap$:
$\mathcal{A} \mathbin{\&} \mathcal{B} \multimap \mathcal{C}$ is not the type of bilinear functions.

$\rightsquigarrow$ Do you see why?



tensor

Tensor product

Definition

The tensor product of $\mathcal{A}$ and $\mathcal{B}$, denoted $\mathcal{A} \otimes \mathcal{B}$ has web $|\mathcal{A}| \times |\mathcal{B}|$, and coherence:

$$(\alpha, \beta) \frown_{\mathcal{A} \otimes \mathcal{B}} (\alpha', \beta') \iff (\alpha \frown_{\mathcal{A}} \alpha' \wedge \beta \frown_{\mathcal{B}} \beta')$$

Tensor product

Definition

The tensor product of $\mathcal{A}$ and $\mathcal{B}$, denoted $\mathcal{A} \otimes \mathcal{B}$ has web $|\mathcal{A}| \times |\mathcal{B}|$, and coherence:

$$(\alpha, \beta) \frown_{\mathcal{A} \otimes \mathcal{B}} (\alpha', \beta') \iff (\alpha \frown_{\mathcal{A}} \alpha' \wedge \beta \frown_{\mathcal{B}} \beta')$$

Then $\mathcal{A} \multimap (\mathcal{B} \multimap \mathcal{C})$ is isomorphic to $(\mathcal{A} \otimes \mathcal{B}) \multimap \mathcal{C}$.

$\rightsquigarrow$ Write down the isomorphism.

Tensor product

Definition

The tensor product of $\mathcal{A}$ and $\mathcal{B}$, denoted $\mathcal{A} \otimes \mathcal{B}$ has web $|\mathcal{A}| \times |\mathcal{B}|$, and coherence:

$$(\alpha, \beta) \frown_{\mathcal{A}\otimes\mathcal{B}} (\alpha', \beta') \iff (\alpha \frown_{\mathcal{A}} \alpha' \wedge \beta \frown_{\mathcal{B}} \beta')$$

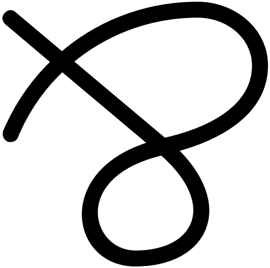
Then $\mathcal{A} \multimap (\mathcal{B} \multimap \mathcal{C})$ is isomorphic to $(\mathcal{A} \otimes \mathcal{B}) \multimap \mathcal{C}$.

$\rightsquigarrow$ Write down the isomorphism.

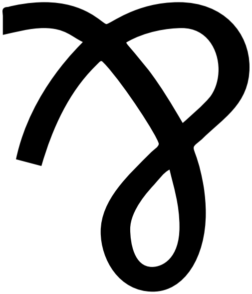
This product is associative and symmetric ($\mathcal{A} \otimes \mathcal{B} \cong \mathcal{B} \otimes \mathcal{A}$), and has a unit:

Definition

Let $\mathbf{1}$ be the only coherence space with web $\{\emptyset\}$.



par



par

Multiplicatives

By de Morgan, we obtain the connective $⅋$ as dual to $\otimes$:

$$\mathcal{A} ⅋ \mathcal{B} = (\mathcal{A}^\perp \otimes \mathcal{B}^\perp)^\perp$$

and then

$$\mathcal{A} \multimap \mathcal{B} = \mathcal{A}^\perp ⅋ \mathcal{B}$$

$\leadsto$ Write down an explicit definition of coherence for $⅋$

Multiplicatives

By de Morgan, we obtain the connective $\parr$ as dual to $\otimes$:

$$\mathcal{A} \parr \mathcal{B} = (\mathcal{A}^\perp \otimes \mathcal{B}^\perp)^\perp$$

and then

$$\mathcal{A} \multimap \mathcal{B} = \mathcal{A}^\perp \parr \mathcal{B}$$

$\leadsto$ Write down an explicit definition of coherence for $\parr$

We thus have three connectives based on set theoretical product, aka. multiplicatives:

- linear implication: $\multimap$;
- the associated conjunction: $\otimes$;
- the associated disjunction: $\parr$.

Multiplicatives

By de Morgan, we obtain the connective $\parr$ as dual to $\otimes$:

$$\mathcal{A} \parr \mathcal{B} = (\mathcal{A}^{\perp} \otimes \mathcal{B}^{\perp})^{\perp}$$

and then

$$\mathcal{A} \multimap \mathcal{B} = \mathcal{A}^{\perp} \parr \mathcal{B}$$

$\rightsquigarrow$ Write down an explicit definition of coherence for $\parr$

We thus have three connectives based on set theoretical product, aka. multiplicatives:

- linear implication: $\multimap$;
- the associated conjunction: $\otimes$;
- the associated disjunction: $\parr$.

The unit of $\parr$ is $\perp = \mathbf{1}^{\perp}$.

A (minor) degeneracy in this case: $\perp = \mathbf{1}$

Sequents

Linear negation is classical (because it is involutive).

Sequents

Linear negation is classical (because it is involutive).

The framework of choice is thus that of sequents: a proof of

$$\mathcal{A}_1, \ldots, \mathcal{A}_n \vdash \mathcal{B}_1, \ldots, \mathcal{B}_p$$

denotes a linear map

$$\mathcal{A}_1 \otimes \ldots \otimes \mathcal{A}_n \multimap \mathcal{B}_1 \parr \ldots \parr \mathcal{B}_p$$

and negation swaps sides

$$(\mathcal{A} \vdash \mathcal{B}) \cong (\mathcal{B}^\perp \vdash \mathcal{A}^\perp)$$

so we can freely move things around.

Linear sequents

Recall that...

The product & is not adjoint to $\multimap$. Hence, in the process of proving, we can't:

- duplicate hypotheses (there is no diagonal for $\otimes$)
- discard hypotheses (the terminal object is not $\mathbf{1}$)

In other words, the structural rules of contraction and weakening do not hold.

Multiplicative linear logic

identity (axiom) $$\frac{}{A \vdash A}$$

composition (cut) $$\frac{\Gamma \vdash A \qquad A \vdash \Delta}{\Gamma \vdash \Delta}$$

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

unit $$\frac{}{\vdash \mathbf{1}}$$

par $$\frac{\Gamma \vdash A, B}{\Gamma \vdash A \parr B}$$

bottom $$\frac{\Gamma \vdash}{\Gamma \vdash \bot}$$

$$\&$$

direct product, with

Product

Recall that…

The direct product $\mathcal{A}_1 \,\&\, \mathcal{A}_2$ has web $|\mathcal{A}_1| + |\mathcal{A}_2|$ and coherence:

$$(i, \alpha) \frown_{\mathcal{A}_1 \& \mathcal{A}_2} (j, \alpha') \iff (i = j \Rightarrow \alpha \frown_{\mathcal{A}_i} \alpha')$$

- It is the type of pairs:
  a clique in $\mathcal{A}_1 \,\&\, \mathcal{A}_2$ can be uniquely written as $a_1 + a_2$.
- Its unit is $\top$: the only space with empty web.

Product

Recall that...

The direct product $\mathcal{A}_1 \,\&\, \mathcal{A}_2$ has web $|\mathcal{A}_1| + |\mathcal{A}_2|$ and coherence:

$$(i, \alpha) \frown_{\mathcal{A}_1 \& \mathcal{A}_2} (j, \alpha') \iff (i = j \Rightarrow \alpha \frown_{\mathcal{A}_i} \alpha')$$

- It is the type of pairs:
  a clique in $\mathcal{A}_1 \,\&\, \mathcal{A}_2$ can be uniquely written as $a_1 + a_2$.
- Its unit is $\top$: the only space with empty web.

pairing (with) $\dfrac{\Gamma \vdash A \qquad \Gamma \vdash B}{\Gamma \vdash A \,\&\, B}$

terminal object (top) $\dfrac{}{\Gamma \vdash \top}$

$$\oplus$$

direct sum, plus

Sums

The category $\mathbf{Coh}_l$ also has coproducts (or sums), dual to &:

Definition

The direct sum $\mathcal{A}_1 \oplus \mathcal{A}_2$ has web $|\mathcal{A}_1| + |\mathcal{A}_2|$ and coherence:

$$(i, \alpha) \frown_{\mathcal{A}_1 \oplus \mathcal{A}_2} (j, \alpha') \iff (i = j \wedge \alpha \frown_{\mathcal{A}_i} \alpha')$$

The unit is $\mathbf{0} = \top^{\perp}$.

Again: $\mathbf{0} = \top$

Sums

The category $\mathbf{Coh}_l$ also has coproducts (or sums), dual to &:

Definition

The direct sum $\mathcal{A}_1 \oplus \mathcal{A}_2$ has web $|\mathcal{A}_1| + |\mathcal{A}_2|$ and coherence:

$$(i, \alpha) \frown_{\mathcal{A}_1 \oplus \mathcal{A}_2} (j, \alpha') \iff (i = j \wedge \alpha \frown_{\mathcal{A}_i} \alpha')$$

The unit is $\mathbf{0} = \top^\perp$.

Again: $\mathbf{0} = \top$

injections (plus) $\dfrac{\Gamma \vdash B}{\Gamma \vdash A \oplus B}$ $\dfrac{\Gamma \vdash A}{\Gamma \vdash A \oplus B}$

Sums

The category $\mathbf{Coh}_l$ also has coproducts (or sums), dual to &:

Definition

The direct sum $\mathcal{A}_1 \oplus \mathcal{A}_2$ has web $|\mathcal{A}_1| + |\mathcal{A}_2|$ and coherence:

$$(i, \alpha) \frown_{\mathcal{A}_1 \oplus \mathcal{A}_2} (j, \alpha') \iff (i = j \wedge \alpha \frown_{\mathcal{A}_i} \alpha')$$

The unit is $\mathbf{0} = \top^{\perp}$.

Again: $\mathbf{0} = \top$

injections (plus) $\dfrac{\Gamma \vdash B}{\Gamma \vdash A \oplus B}$ $\dfrac{\Gamma \vdash A}{\Gamma \vdash A \oplus B}$

By contrast, there was no sum type in the stable semantics!

Additives

Together & and $\oplus$ are the additive connectives of linear logic.
Moreover:

$$\mathcal{A} \otimes (\mathcal{B} \oplus \mathcal{C}) \cong (\mathcal{A} \otimes \mathcal{B}) \oplus (\mathcal{A} \otimes \mathcal{C})$$

(and dually for $\invamp$ and &).

!?

"of course" and "why not": the exponentials

Exponentials

Recall that...

The space $!\mathcal{A}$ has web $\mathcal{A}_{\text{fin}}$ and coherence:

$$a \frown_{!\mathcal{A}} a' \iff a \cup a' \in \mathcal{A}.$$

Its de Morgan dual is denoted $?A = (!A^{\perp})^{\perp}$.

Structural rules hold on exponentials

$$\text{contraction} \quad \frac{!A, \qquad !A \vdash \Delta}{!A \vdash \Delta}$$

$$\text{weakening} \quad \frac{\vdash \Delta}{!A \vdash \Delta}$$

Exponentials

Recall that...

The space $!\mathcal{A}$ has web $\mathcal{A}_{\text{fin}}$ and coherence:

$$a \frown_{!\mathcal{A}} a' \iff a \cup a' \in \mathcal{A}.$$

Its de Morgan dual is denoted $?A = (!A^{\perp})^{\perp}$.

Structural rules hold on exponentials

contraction $\dfrac{a : !A, a' : !A \vdash \Delta}{a \cup a' : !A \vdash \Delta}$

weakening $\dfrac{\vdash \Delta}{\emptyset : !A \vdash \Delta}$

Introduction rules for exponentials: dereliction

Linearization of the (stable) identity $\lambda x\, x : \mathcal{A} \to \mathcal{A}$ gives

$$\{(\{\alpha\}\,,\alpha);\ \alpha \in |\mathcal{A}|\} \in {!\mathcal{A}} \multimap \mathcal{A}$$

Hence the rule

$$\frac{A \vdash \Delta}{!A \vdash \Delta} \quad \text{or equivalently} \quad \frac{\vdash A, \Delta}{\vdash ?A, \Delta}$$

Introduction rules for exponentials: promotion

Exponentials are functors

If $f : \mathcal{A} \multimap \mathcal{B}$, define $!f : !\mathcal{A} \multimap !\mathcal{B}$ by its trace

$$\{(\{\alpha_1, \ldots, \alpha_n\}, \{\beta_1, \ldots, \beta_n\}) \in \mathcal{A}_{\text{fin}} \times \mathcal{B}_{\text{fin}};\ \forall i,\ (\alpha_i, \beta_i) \in f\}$$

So we could have a rule

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

Introduction rules for exponentials: promotion

Exponentials are (co-)monads

We only miss the "multiplication" $!\mathcal{A} \multimap !!\mathcal{A}$, given by

$$\left\{ \left(\bigcup_{j=1}^{n} a_j, \{a_1, \ldots, a_n\} \right) \in \mathcal{A}_{\mathrm{fin}} \times !\mathcal{A}_{\mathrm{fin}} \right\}$$

which would give a rule

$$\frac{!!A \vdash \Delta}{!A \vdash \Delta}$$

called digging.

Introduction rules for exponentials: promotion

The previous two are subsumed by the promotion rule:

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

Introduction rules for exponentials: promotion

The previous two are subsumed by the promotion rule:

$$\frac{(!\Gamma_j \vdash \alpha_j : A)_{j=1}^{k}}{\bigcup_{j=1}^{k} !\Gamma_j \vdash \{\alpha_1, \ldots, \alpha_k\} : !A}$$

Cut elimination

- Next talk!
- Reflects properties of the things we used (properties of functors, products, sums, ...).
- Makes the system logically relevant.

Cut elimination

- Next talk!
- Reflects properties of the things we used (properties of functors, products, sums, . . .).
- Makes the system logically relevant.
- Refines β-reduction. . .

Lambda-calculus in LL

Principle

The decomposition $A \to B = !A \multimap B$ leads to translate a typed term $\Gamma \vdash s : A$ as a proof of $!\Gamma \vdash A$.

Variable

$$\frac{}{\Delta^{\emptyset}, x^{\{\alpha\}} : A \vdash x^{\alpha} : A}$$

becomes

$$\dfrac{\dfrac{\dfrac{}{\alpha : A \vdash \alpha : A}}{\{\alpha\} : !A \vdash \alpha : A}\ (\mathrm{der})}{\emptyset : !\Delta, \{\alpha\} : !A \vdash \alpha : A}\ (\mathrm{weak})^*$$

Lambda-calculus in LL

Abstraction

$$\frac{\Gamma, x^{a} : A \vdash s^{\beta} : B}{\Gamma \vdash \lambda x\, s^{(a,\beta)} : A \to B}$$

becomes

$$\frac{!\Gamma, a : !A \vdash \beta : B}{!\Gamma \vdash (a, \beta) : !A \multimap B}\ (\multimap)$$

Lambda-calculus in LL

Application

$$\dfrac{\Gamma_0 \vdash s^{(\{\alpha_1,\dots,\alpha_k\},\beta)} : A \to B \qquad (\Gamma_j \vdash t^{\alpha_j} : A)_{j=1}^{k}}{\bigcup_{j=0}^{k}\Gamma_j \vdash (s)\, t^{\beta} : B}$$

becomes

$$\dfrac{\dfrac{\begin{array}{c} !\Gamma_0 \vdash (\{\alpha_1,\dots,\alpha_k\},\beta) : !A \multimap B \\ \vdots \\ \vdots \\ \vdots \end{array} \qquad \dfrac{\dfrac{(!\Gamma_j \vdash \alpha_j : A)_{j=1}^{k}}{\bigcup_{j=1}^{k} !\Gamma_j \vdash \{\alpha_1,\dots,\alpha_k\} : !A}\,(\text{prom}) \qquad \dfrac{}{\beta : B \vdash \beta : B}\,(\text{ax})}{\bigcup_{j=1}^{k} !\Gamma_j, (\{\alpha_1,\dots,\alpha_k\},\beta) : !A \multimap B \vdash \beta : B}\,(\multimap_l)}{\Gamma_0, \bigcup_{j=1}^{k} !\Gamma_j \vdash \beta : B}\,(\text{cut})}{\bigcup_{j=0}^{k}\Gamma_j \vdash \beta : B}\,(\text{cont})^*$$

Digression 1: Orthogonality

Coherence spaces: a third definition

- If $a, a' \subseteq A$, write $a \perp a'$ iff $a \cap a'$ is at most a singleton.
- If $\mathcal{A} \subseteq \mathcal{P}(A)$, write $\mathcal{A}^\perp = \{a' \subset A;\ \forall a \in \mathcal{A},\ a \perp a'\}$.
- Coherence spaces of web A are those $\mathcal{A}$ such that $\mathcal{A} = \mathcal{A}^{\perp\perp}$.

Then a relation $f \subseteq |\mathcal{A}| \times |\mathcal{B}|$ is a linear trace iff

$$\forall a \in \mathcal{A},\ \forall b' \in \mathcal{B}^\perp,\ f \cdot a \perp b' \wedge a \perp f^\perp \cdot b'$$

Such orthogonality constructions are very common in the design of LL models.

Digression 2: Quantitative semantics

Why do we use such linear algebraic vocabulary, notations and concepts? Historically:

coh. spaces $\Lleftarrow$ qualitative domains $\Lleftarrow$ quantitative semantics

Digression 2: Quantitative semantics

Why do we use such linear algebraic vocabulary, notations and concepts? Historically:

coh. spaces $\Lleftarrow$ qualitative domains $\Lleftarrow$ quantitative semantics

Rough intuition

Interpret terms as a linear combinations: $s = \sum_{\alpha \in s} s_\alpha \alpha$
so that application is given by a power series:

$$((s)\,t)_\beta = \sum_{(a,\beta)\in s} s_{(a,\beta)} t^a$$

where $t^{[\alpha_1,\ldots,\alpha_k]} = t_{\alpha_1} \cdots t_{\alpha_k}$.

Digression 2: Quantitative semantics

Why do we use such linear algebraic vocabulary, notations and concepts? Historically:

coh. spaces $\Lleftarrow$ qualitative domains $\Lleftarrow$ quantitative semantics

Rough intuition

Interpret terms as a linear combinations: $s = \sum_{\alpha \in s} s_\alpha \alpha$
so that application is given by a power series:

$$((s)\,t)_\beta = \sum_{(a,\beta)\in s} s_{(a,\beta)} t^a$$

where $t^{[\alpha_1,\ldots,\alpha_k]} = t_{\alpha_1} \cdots t_{\alpha_k}$.

Taken litterally, it is only meaningful if we can ensure a form of convergence.

The end

Thanks.

The end

Thanks.
Questions?

The end

Thanks.

Questions?

Next talk, Emmanuel Beffara:

sequent calculus, polarities, focalization, phase semantics