Around finite semantics for second-order multiplicative-additive linear logic

Nguyễn Lê Thành Dũng (a.k.a. Tito) — nltd@nguyentito.eu
Laboratoire d'informatique de Paris Nord (LIPN), Université Paris 13
Rencontre mensuelle Chocola, Lyon, mercredi 12 décembre 2018

Based on joint work with:

- Thomas Seiller (CNRS, LIPN), my PhD advisor
- Paolo Pistone (Univ. Tübingen), **next speaker!**
- Lorenzo Tortora de Falco (Univ. Roma Tre)

Theorem

Second order Multiplicative-Additive Linear Logic (MALL2) *admits a finite denotational semantics.*

1. Motivation and applications (intro)
2. Multiple constructions of finite models of MALL2
3. Some vague intuitions and speculation

Why finite semantics?

The simply-typed λ-calculus admits finite denotational models, e.g. FinSet. Applications:

- *semantic evaluation* technique,
  often useful for studying complexity in STλ
- finite semantics for λY (i.e. STλ + fixpoints) lead to semantic approach to *higher-order model checking* (Aehlig, Salvati–Walukiewicz...)

Why finite semantics?

The simply-typed λ-calculus admits finite denotational models, e.g. FinSet. Applications:

- *semantic evaluation* technique, often useful for studying complexity in STλ
- finite semantics for λY (i.e. STλ + fixpoints) lead to semantic approach to *higher-order model checking* (Aehlig, Salvati–Walukiewicz...)

Propositional linear logic admits finite models, e.g. coherence spaces, Scott model. Yields refinements of the above, e.g.:

- Terui, RTA'12: semantic evaluation in the Scott model
  Slogan: "better semantics, faster computation"
- Grellois–Melliès: HOMC with LL semantics

Why finite semantics? An example

Church encoding of bitstrings:
$\mathsf{Str}[A] = (A \to A) \to (A \to A) \to (A \to A)$.

Theorem (Hillebrand & Kanellakis, LICS'96)

The languages decided by STλ*-terms of type* $\mathsf{Str}[A] \to \mathsf{Bool}$ *are exactly the* regular *languages.*

Proof idea: build a DFA whose states are $[\![\mathsf{Str}[A]]\!]$.

- an example of the semantic evaluation technique (indeed, implicit characterization of regular languages)
- a correspondence Church encodings / finite automata; generalization to infinite trees $\to$ HO model checking

Finite semantics for polymorphism

Previous examples work for *monomorphic* type systems.

This seems impossible with impredicative polymorphism.
For instance in System F:

Proposition

Any non-trivial semantics must be injective on $\forall X.\,\mathsf{Str}[X]$.

Finite semantics for polymorphism

Previous examples work for *monomorphic* type systems.

This seems impossible with impredicative polymorphism.
For instance in System F:

Proposition

Any non-trivial semantics must be injective on $\forall X.\, \mathsf{Str}[X]$.

This example relies fundamentally on *non-linearity*.

Actually, there are finite models:

- with impredicative quantification;
- with exponentials;
- but not with both!

Applications for MALL2

Theorem

MALL2 *has a non-trivial finite semantics.*

Consequences on the expressive power of *recursive types*:

- μMALL cannot be embedded in MALL2, since
  infinite datatypes, e.g. initial algebras, cannot be encoded
- characterization of regular languages in second-order Elementary Linear Logic
  - Baillot's characterization of P in ELL, minus type fixpoints
  - my initial motivation for this work!

Towards a syntactic model

Remark: in propositional MALL (MALL0), each formula has finitely many cut-free proofs.
$\rightarrow$ Hope for a *syntactic model* of MALL2, injective on MALL0.

Difficulty of 2nd order case: arbitrarily large $\exists$ witnesses.

Towards a syntactic model

Remark: in propositional MALL (MALL0), each formula has finitely many cut-free proofs.
$\rightarrow$ Hope for a *syntactic model* of MALL2, injective on MALL0.

Difficulty of 2nd order case: arbitrarily large $\exists$ witnesses.
Solution: *observational quotient* of the syntax.

- Choose observations which "cannot inspect witnesses"
- Intuition from programming languages: $\exists$ = abstract types, dual to $\forall$ = generic programs

Equivalence for propositional observations

Definition

Let A be a MALL2 formula and $\pi, \pi' : A$. Define $\pi \sim_A \pi'$ as: for any *propositional* MALL formula B, for any proof ρ of $A \vdash B$, $\mathbf{cut}(\pi, \rho)$ and $\mathbf{cut}(\pi', \rho)$ have the same normal form.

- $\sim$ is a congruence: the quotient is a model of MALL2
- A existential-free $\Rightarrow \sim_A$ trivial
- Example: the proofs of $\exists X.\, X$ cannot be distinguished
    - 2nd-order encodings of units work, e.g. $\top \equiv \exists X.\, X$.

Theorem

For any MALL2 *formula A, there are* finitely many *classes for $\sim_A$.*

Proved with *proof nets* for MLL2, extended to MALL2 by a trick.

The observational quotient is non-effective

The observational quotient seems very concrete and simple.

However, given a type A, one cannot enumerate $A/\sim_A$, even though it is finite. Indeed, one cannot check its emptiness:

Theorem (Lafont 1996)

MALL2 *is undecidable.*

More: adapting Lafont's proof gives (caveat: to be checked)

Proposition

Given a MALL2 *type* A *and* $\pi, \pi' : A$, $\pi \sim_A \pi'$ *is undecidable.*

(However, for a *fixed* A, $\sim_A$ is decidable.)

$\longrightarrow$ Search for *effective* finite models.
To overcome undecidability, enlarge the semantics.

A non-syntactic model: coherence spaces

Let's come back to the origins of linear logic
$\rightarrow$ decomposition $A \Rightarrow B \equiv\ !A \multimap B$ in *coherence spaces*
$\rightarrow$ "The system F of variable types, fifteen years later"
Girard's model of impredicative polymorphism

Usual "selling point" of coherence spaces (esp. w.r.t. Scott domains): *small* and legible interpretations of types.

Actually, they are even finite and effective for MALL2!

A non-syntactic model: coherence spaces

Let's come back to the origins of linear logic
$\rightarrow$ decomposition $A \Rightarrow B \equiv\ !A \multimap B$ in *coherence spaces*
$\rightarrow$ "The system F of variable types, fifteen years later"
Girard's model of impredicative polymorphism

Usual "selling point" of coherence spaces (esp. w.r.t. Scott domains): *small* and legible interpretations of types.

Actually, they are even finite and effective for MALL2!

What about the *relational* model?
Simpler, but not quite a model of MALL2...
(issue investigated by A. Bac, T. Ehrhard and C. Tasson)
But let's start with that for pedagogical purposes.

A relational example

Let $\pi : X^{\perp} \parr X$. For all sets S, $[\![\pi]\!]_{X \mapsto S} \subseteq S \times S$ in Rel.

This family should somehow be "uniform" in S, such that

$$\text{uniform families } r_S \subset S \times S \cong \text{subsets } r \subset \left[\!\!\left[\forall X.\, X^{\perp} \parr X \right]\!\!\right]$$

to interpret the $\forall$-intro rule.

A relational example

Let $\pi : X^{\perp} \parr X$. For all sets S, $[\![\pi]\!]_{X \mapsto S} \subseteq S \times S$ in Rel.

This family should somehow be "uniform" in S, such that

$$\text{uniform families } r_S \subset S \times S \cong \text{subsets } r \subset \left[\!\!\left[\forall X.\, X^{\perp} \parr X\right]\!\!\right]$$

to interpret the $\forall$-intro rule.

What makes sense in all sets: $=$, $\neq$.
Use "bound variables" to represent equality. Binder $\langle x, y, \ldots \rangle$.

$$\begin{aligned}
\{(s,s) \mid s \in S\} &\leftrightarrow \{\langle x \rangle (x,x)\} \\
\{(s,s') \mid s \neq s' \in S\} &\leftrightarrow \{\langle x,y \rangle (x,y)\} \\
S \times S &\leftrightarrow \{\langle x \rangle (x,x), \langle x,y \rangle (x,y)\}
\end{aligned}$$

A relational example

Let $\pi : X^{\perp} ⅋ X$. For all sets S, $[\![\pi]\!]_{X \mapsto S} \subseteq S \times S$ in Rel.

This family should somehow be "uniform" in S, such that

$$\text{uniform families } r_S \subset S \times S \cong \text{subsets } r \subset \left[\!\!\left[\forall X.\, X^{\perp} ⅋ X\right]\!\!\right]$$

to interpret the $\forall$-intro rule.

What makes sense in all sets: $=$, $\neq$.
Use "bound variables" to represent equality. Binder $\langle x, y, \ldots \rangle$.

$$\{(s, s) \mid s \in S\} \leftrightarrow \{\langle x \rangle (x, x)\}$$

$$\{(s, s) \mid s \in S\} = \{(\iota(x), \iota(x)) \mid \iota : \{x\} \hookrightarrow S\}$$

$$\{(s, s') \mid s \neq s' \in S\} \leftrightarrow \{\langle x, y \rangle (x, y)\}$$

$$\{(s, s') \mid s \neq s' \in S\} = \{(\iota(x), \iota(y)) \mid \iota : \{x, y\} \hookrightarrow S\}$$

Normal functors

In general: functor $F : \mathsf{Inj} \to \mathsf{Inj}$ (category of injections), representing open type with one variable.

Representation of $\forall$: $\mathrm{Tr}(F)$ consists of pairs $\langle X \rangle x$ with $x \in F(X)$.
We require X *minimal*: $X' \subsetneq X \implies x \notin F(X')$.

Existence of *unique minimal* X for each given x:
ensured for functors *preserving pullbacks* ($\simeq$ intersections) (1).
And if F preserves *filtered colimits* ($\simeq$ directed unions) (2),
the minimal X is *finite*.

(Girard's "normal form theorem"; $\langle X \rangle x$ is called a *normal form*.
(1)+(2) = definition of *normal functors* (or stable functors).)

Normal functors of finite degree

New: for our purposes, we consider functors:

- preserving finiteness: $\mathrm{Card}(X) < \infty \implies \mathrm{Card}(F(X)) < \infty$
- of *finite degree*: $\deg(F) = \sup\{\mathrm{Card}(X) \mid \langle X \rangle x \in \mathrm{Tr}(F)\} < \infty$

(Mutatis mutandis for open types with n variables.)

Intuition: $\deg(F)$ = max number of bound variables used
= number of loci where an $x \in X$ may appear in a point of $F(X)$

$$\deg(F^\perp) = \deg(F) \quad \deg(X \mapsto \mathrm{Tr}(F(X, -))) \leq \deg(F)$$

$$\deg(F \otimes G) \leq \deg(F) + \deg(G) \quad \deg(F \oplus G) \leq \max(\deg(F), \deg(G))$$

Also, $\mathrm{Card}(\mathrm{Tr}(F(X, -))$ bounded using $\deg(F)$.

$\longrightarrow$ finite (non-)model of MALL2, obviously effective

The relational model is not a model

An issue with composition: take

- $\langle x \rangle(*, x) \leftrightarrow \{*\} \times S \subseteq [\![1 \multimap X]\!]_{X \mapsto S}$
- $\langle x \rangle(x, *) \leftrightarrow S \times \{*\} \subseteq [\![X \multimap 1]\!]_{X \mapsto S}$

Compose them to get...

The relational model is not a model

An issue with composition: take

- $\langle x \rangle (*, x) \leftrightarrow \{*\} \times S \subseteq [\![1 \multimap X]\!]_{X \mapsto S}$
- $\langle x \rangle (x, *) \leftrightarrow S \times \{*\} \subseteq [\![X \multimap 1]\!]_{X \mapsto S}$

Compose them to get... $\{(*, *)\}$ if $S \neq \emptyset$, $\emptyset$ if $S = \emptyset$.
Clearly not a uniform family for the constant functor $1 \multimap 1$!

The relational model is not a model

An issue with composition: take

- $\langle x \rangle (*, x) \leftrightarrow \{*\} \times S \subseteq [\![1 \multimap X]\!]_{X \mapsto S}$
- $\langle x \rangle (x, *) \leftrightarrow S \times \{*\} \subseteq [\![X \multimap 1]\!]_{X \mapsto S}$

Compose them to get... $\{(*, *)\}$ if $S \neq \emptyset$, $\emptyset$ if $S = \emptyset$.
Clearly not a uniform family for the constant functor $1 \multimap 1$!

Yet, composition works for the coherence space model of polymorphism; miraculous consequence of *stability*
(i.e. coherence space morphisms $\simeq$ lower-dim normal functors)
Issue overlooked by Girard, solved by "Moggi's trick"
(mentioned in *Proofs and Types* appendix for another purpose!).

Let's do the same thing in coherence spaces!

We consider the category CohI:

- objects: coherence spaces (graphs X over vertex set $|X|$)
- morphisms $X \hookrightarrow Y$: *embeddings*, i.e. a graph isomorphism between X and an induced subgraph of Y

Open type with n type variables $\mapsto$ functor $F : \mathsf{CohI}^n \to \mathsf{CohI}$.
For instance "$\multimap$" is indeed *covariant in both variables*:
if $X \subseteq X'$, $Y \subseteq Y'$, then $X \multimap Y \subseteq X' \multimap Y'$.

Again, we require F to be a *normal functor*,
i.e. preserving filtered colimits and pullbacks in CohI.

This leads to a correspondence:

$$\text{uniform families } c_X \in \text{Cliques}(F(X)) \;\cong\; c \in \text{Cliques}(\text{Tr}(F))$$

Coherent model vs. relational model

Now $\mathrm{Tr}(F)$ has a more involved definition:
we consider normal forms $\langle X\rangle x$ – X being a coherence space – and exclude the "self-incoherent" ones from the web $|\mathrm{Tr}(F)|$.
$\rightarrow$ a sort of "correctness criterion"

Revisiting previous examples:

- $[\![\forall X.\, X^{\perp} \parr X]\!] = \{\langle x\rangle(x,x)\}$ – only 1 self-coherent point
- $[\![\forall X.\, 1 \multimap X]\!] = [\![\forall X.\, X \multimap 1]\!] = \emptyset\}$
  thus no problem with composition here!

New intuition for $\langle X\rangle$: set of bound variables *with coherence*, what makes sense in all coherence spaces is $=$, $\neq$, $\frown$, $\smile$.
Inj: preserve $=$, $\neq$. CohI: also preserve $\frown$, $\smile$.

Coherence spaces: the model we were looking for

Need to handle coherence relation everywhere:
cumbersome, but still effectively computable.

$\deg(-)$ and $\mathrm{Card}(\mathrm{Tr}(-))$ have the same upper bounds as before.

Theorem

Coherence spaces provide an effective finite semantics for MALL2*.*

This semantics is even *injective* for MALL0
$\rightarrow$ proves finiteness of observational quotient!

So what is happening here?

A general remark on ∃ and cut

In coherence spaces, ∀-elim easily defined by:

$$\{(\langle X\rangle x, F(\iota)(x)) \mid \langle X\rangle x \in \mathrm{Tr}(F),\ \iota : X \hookrightarrow S\} \subseteq (\mathrm{Tr}(F) \multimap F(S))$$

∃-intro less primitive, obtained by transposing the above.
Non-trivial computational contents, *erasing witnesses*:
e.g. subsumes cut, since $[\![\exists X.\, X \otimes X^\perp]\!] = [\![\perp]\!]$

$$\frac{\vdash \Gamma, A \quad \vdash \Delta, A^\perp}{\vdash \Gamma, \Delta} \quad \sim \quad \frac{\vdash \Gamma, \Delta, A \otimes A^\perp}{\vdash \Gamma, \Delta, \exists X.\, X \otimes X^\perp}$$

(as already remarked by Girard in 1986)

In general, ∃-intro precomputes reaction to all possible ∀s
→ allows compression of information, thus finiteness.

A syntactic approach of observational equivalence

This intuition also appears in a *purely syntactic* finiteness proof for the observational quotient. Reminder:

Definition

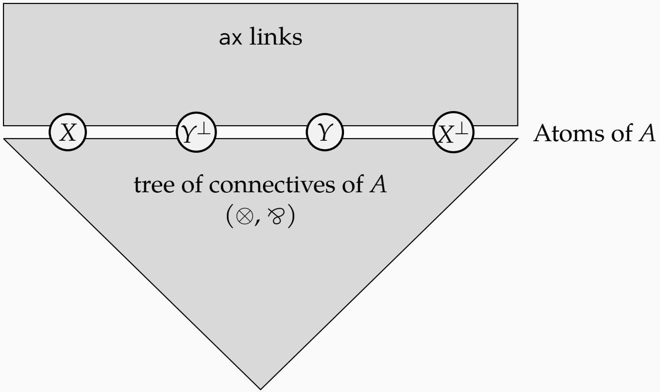
Let A be a MALL2 formula and $\pi, \pi' : A$. Define $\pi \sim_A \pi'$ as: for any *propositional* MALL formula B, for any proof ρ of $A \vdash B$, $\mathbf{cut}(\pi, \rho)$ and $\mathbf{cut}(\pi', \rho)$ have the same normal form.

Theorem

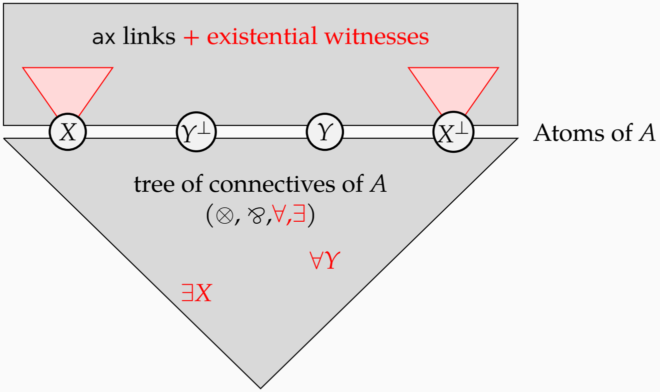
For any MALL2 *formula* A*, there are* finitely many *classes for* $\sim_A$.

Next slides: familiarity with proof nets is assumed.

What a propositional MLL proof net looks like

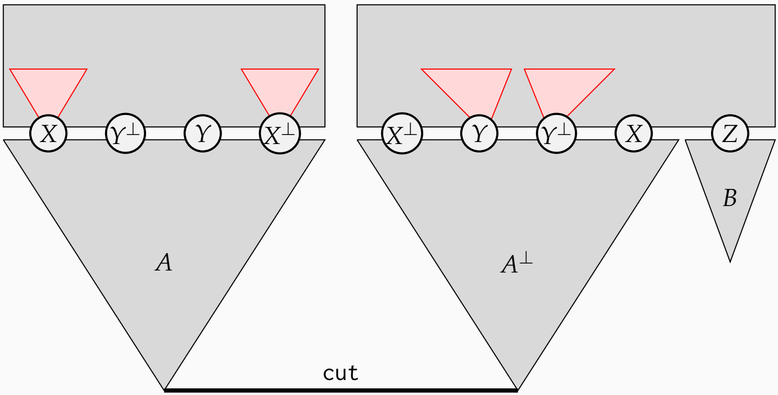


What a MLL2 proof net looks like



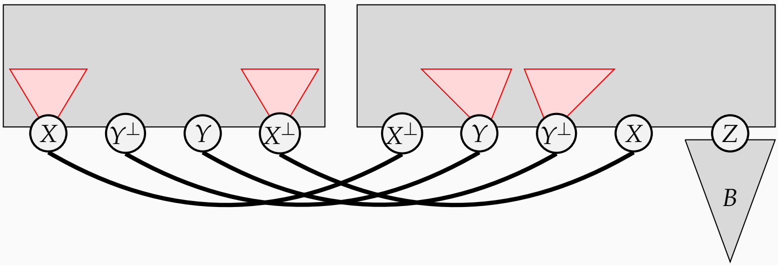
Finiteness theorem (1): proof/observation interaction

A: MLL2 formula; B: propositional MLL formula

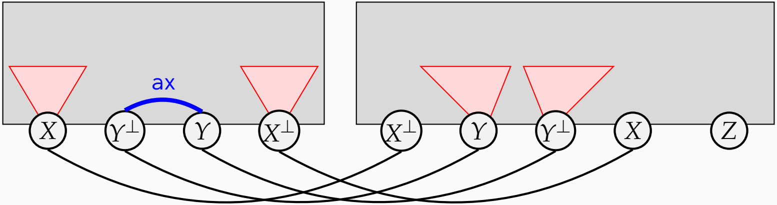


Finiteness theorem (1): proof/observation interaction

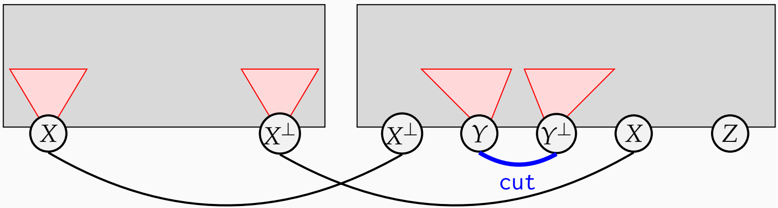
A: MLL2 formula; B: propositional MLL formula



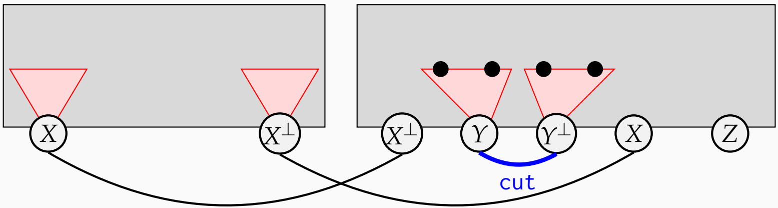
Finiteness theorem (2): eliminating two cuts



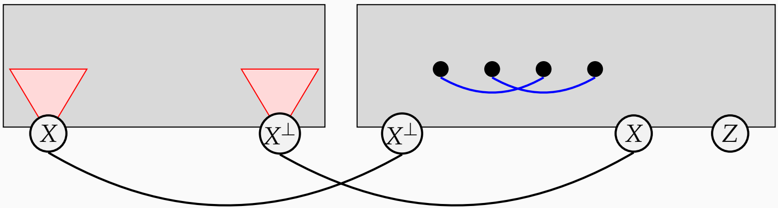
Finiteness theorem (2): eliminating two cuts



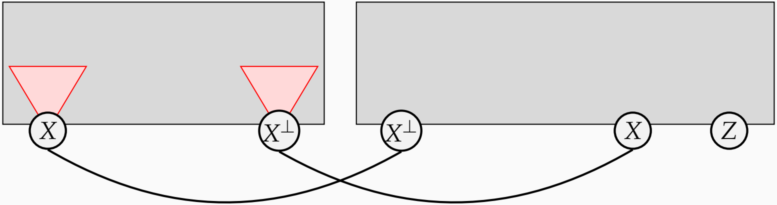
Finiteness theorem (2): eliminating two cuts



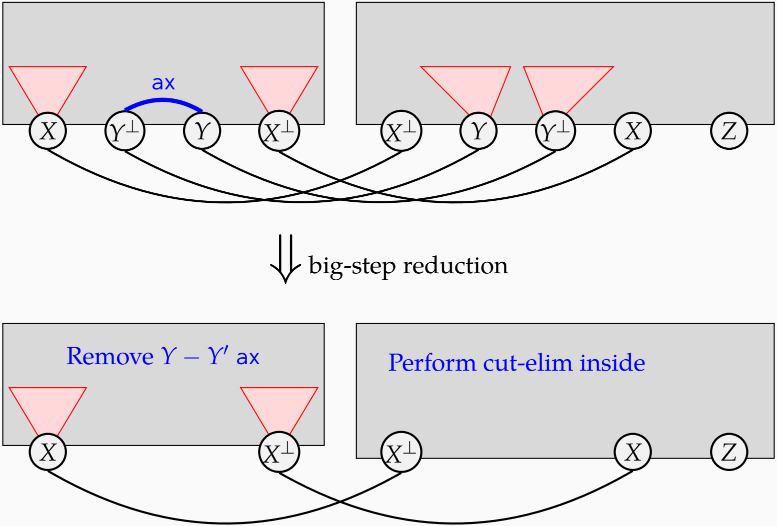
Finiteness theorem (2): eliminating two cuts



Finiteness theorem (2): eliminating two cuts

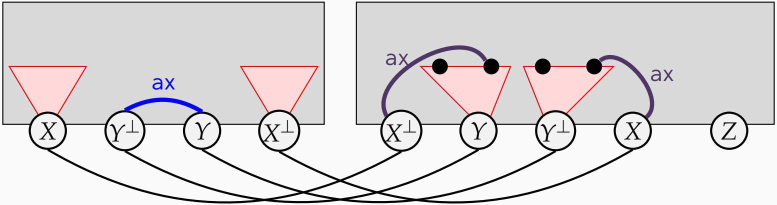


Finiteness theorem (3): big-step reduction



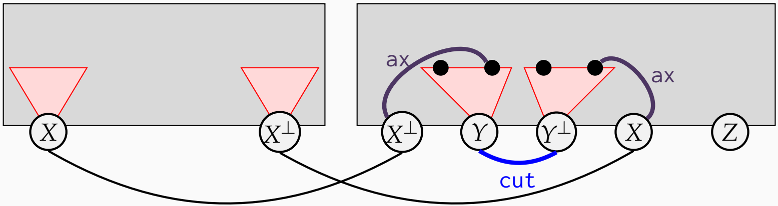
Finiteness theorem (4): normalization

New redexes may appear during reduction.



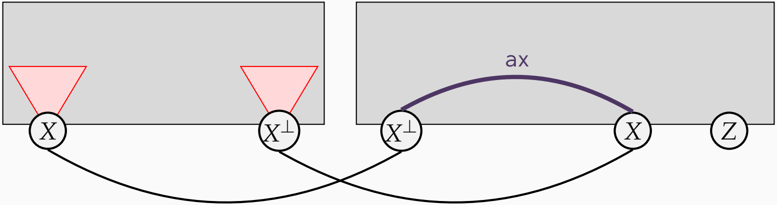
Finiteness theorem (4): normalization

New redexes may appear during reduction.



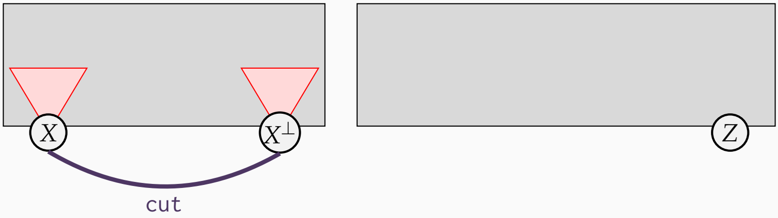
Finiteness theorem (4): normalization

New redexes may appear during reduction.

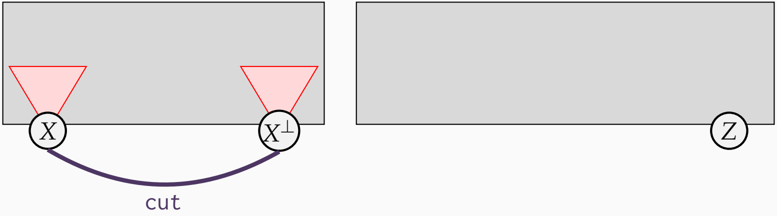


Finiteness theorem (4): normalization

New redexes may appear during reduction.



New redexes may appear during reduction.



Lemma (Progress)

As long as there remains a `cut`*-link, there is a redex for* $\Rightarrow$.

This is the tricky part of the proof.

MLL2 normalization as a dialogue

The reaction of a proof to all possible sequences of $\forall$ axiom links of an observation determines its equivalence class.
As in coherence spaces, witness-free and bounded:
hence the theorem.

New intuition: *dialogue/game* between proof and observation, through the interface of the cut formula.

Can we translate this proof into the construction of a model?
How to define "strategies" which compose?
Game semantics? Geometry of Interaction? Ludics?

An analogy with additives

$\forall/\exists$ interaction is somewhat like $\oplus/\&$ cut. Approximately:

- Proof of $\forall$: plays a choice of ax-links
- Proof of $\exists$: yields one sub-proof for each choice

Remark: *polarities are inverted*, $\forall$ positive, $\exists$ negative.

Suggests something like focusing or ludics...
but the exchange of ax-links seems not orderly enough.
Attempts to prove "progress lemma" by finding structure,
e.g. some "dependency order" of quantifiers, all unsuccessful.

Instead, use a general notion of normalization as feedback
$\rightarrow$ geometry of interaction.

Categorical GoI for MLL

Traced monoidal category $\mathcal{C} \rightsquigarrow$ model $\mathsf{Int}(\mathcal{C})$ of MLL.
(*Trace* = abstract notion of feedback)

- Objects: pairs $A = (A^+, A^-)$; duality: $A^\perp = (A^-, A^+)$
- Morphisms: $\mathrm{Hom}_{\mathsf{Int}(\mathcal{C})}(A, B) = \mathrm{Hom}_{\mathcal{C}}(A^+ \otimes B^-, A^- \otimes B^+)$
- Composition $A \to B \to C$: feedback on B^+ and B^-

Informal dichotomy on monoidal product $\otimes$ of $\mathcal{C}$
(Abramsky, "Retracing some paths in process algebra"):

- "Particle-style" GoI: $\otimes$ coproduct-like
  - related to token machine GoI, Girard's original version, etc.
- "Wave-style" GoI: $\otimes$ cartesian product
  - less common than particle-style
  - e.g. Abramsky–Jagadeesan "New Foundations" GoI

Particle-style vs wave-style GoI

"Particle-style" GoI: *stateless* dialogue (e.g. token-passing)

- benefit: logspace evaluation (store the token only)
- mismatch with additives
- mismatch with our interpretation of quantifiers

"Wave-style" GoI: *history-dependent* dialogue

- trace $\cong$ parameterized fixpoint; intuition:
  possibly unstructured iteration until convergence
- seems to accomodate our quantifiers!
- hope: for restricted situations, sub-polynomial evaluation

The last point could learn to an implicit characterization of logspace in second-order ELL.

Wave-style GoI for MLL2?

Let $\mathcal{C}$ cartesian traced. $\mathsf{Int}(\mathcal{C})$ compact closed, $\otimes = \parr$.

For "nice enough" $\mathcal{C}$ we believe that

$$\forall X.\, F(X) \equiv F(1) \otimes L_F \qquad \exists X.\, F(X) \equiv F(1) \otimes L_F^{\perp}$$

Futhermore L_F should be the form $(L_F^+, 1)$
$\rightarrow$ objects interpreting MALL2 are $(\{\forall$ ax-links$\}, \{\exists$ ax-links$\})$,
whose elements are $\mathcal{C}$-morphisms $\{\exists$ ax-links$\} \rightarrow \{\forall$ ax-links$\}$.

So L_F is *positive*, of the form $(A^+, 1)$.
Justification: positive objects can be duplicated/erased.
(Hasegawa's exponential: $!(A^+, A^-) = (A^- \Rightarrow A^+, 1)$.)

So, again, $\forall$ positive and $\exists$ negative.

Inversion of polarities

$$A \otimes B \equiv \forall X. (A \multimap B \multimap X) \multimap X$$

$$A \oplus B \equiv \forall X. !(A \multimap X) \multimap !(B \multimap X) \multimap X$$

"The second-order translations do invert polarities. [...] *This inversion of polarities is due to deep reasons – so deep that they remain completely obscure."*

— Girard, *The Blind Spot*, §12.B.2