A gentle introduction to Girard’s Transcendental Syntax

LIPN – Université Sorbonne Paris Nord

Boris Eng    Thomas Seiller
A bit of context

*Geometry of Interaction (GoI)*

A lot of definitions...But in our case:
A bit of context

*Geometry of Interaction (GoI)*

A lot of definitions...But in our case:

Girard’s original Geometry of Interaction [GoI I, 1989].
A bit of context

*Geometry of Interaction (GoI)*

A lot of definitions...But in our case:

Girard’s original Geometry of Interaction [GoI I, 1989].

**Goal**: study the dynamics of linear logic from computation (operator algebras).
A bit of context

*Geometry of Interaction (GoI)*

A lot of definitions...But in our case:

Girard’s original *Geometry of Interaction* [GoI I, 1989].

Goal: study the dynamics of linear logic from *computation* (operator algebras).

Transcendental Syntax [GoI VI, 2013]: the successor.
A bit of context

*Geometry of Interaction (GoI)*

A lot of definitions...But in our case:

Girard’s original Geometry of Interaction [GoI I, 1989].

**Goal**: study the *dynamics* of linear logic from *computation* (operator algebras).

Transcendental Syntax [GoI VI, 2013]: the successor.

**Goal**: linear logic (proof-nets) as *emerging* from computation *without semantics*.
A bit of context

*Geometry of Interaction (GoI)*

A lot of definitions... But in our case:

Girard’s original Geometry of Interaction [GoI I, 1989].

- **Goal**: study the **dynamics** of linear logic from **computation** (operator algebras).

Transcendental Syntax [GoI VI, 2013]: the successor.

- **Goal**: linear logic (proof-nets) as **emerging** from computation **without semantics**.

- **Computational bricks**: "stellar resolution" (not the only possibility).
A bit of context

*Geometry of Interaction (GoI)*

A lot of definitions...But in our case:

Girard’s original Geometry of Interaction [GoI I, 1989].

- **Goal**: study the *dynamics* of linear logic from *computation* (operator algebras).


- **Goal**: linear logic (proof-nets) as *emerging* from computation *without semantics*.
- **Computational bricks**: "stellar resolution" (not the only possibility).
- **Logical correctness**: by symmetric computational testing.
Stellar Resolution

Between tilings and logic programming

"Flexible" tiles [stars] with (un)polarised terms [rays]. Group as [constellations].

\[
\phi \frac{g(x)}{\text{one.pnum}} + a(x) - b(x) \quad \phi \frac{a(f(y))}{\text{two.pnum}} - a(f(y)) + c(y)
\]

Evaluation: link-contraction by Robinson’s Resolution rule.

Execution: construct all possible connected & maximal tilings then evaluate them.
Stellar Resolution

Between tilings and logic programming

"Flexible" tiles [stars] with (un)polarised terms [rays]. Group as [constellations].

\[
g(x) \cdot \phi_1 \cdot \phi_2 \cdot \\
\ + a(x) \quad \ldots \quad \ - a(f(y)) \quad \ + c(y) \\
\ - b(x) \cdot \\
\]

Evaluation: link-contraction by Robinson’s Resolution rule.

Execution: construct all possible connected & maximal tilings then evaluate them.
Stellar Resolution

Between tilings and logic programming

"Flexible" tiles [stars] with (un)polarised terms [rays]. Group as [constellations].

\[ g(x) \rightarrow \phi_1 + a(x) \rightarrow -a(f(y)) \rightarrow \phi_2 + c(y) \rightarrow -b(x) \]

Evaluation : link-contraction by Robinson’s Resolution rule.
Stellar Resolution

Between tilings and logic programming

"Flexible" tiles [stars] with (un)polarised terms [rays]. Group as [constellations].

\[
\phi_{\text{one.pnum}} \quad g(x) \cdot \phi_1 \cdot +a(x) \cdot -a(f(y)) \cdot +c(y) \cdot -b(x) \cdot \phi_2
\]

**Evaluation**: link-contraction by Robinson’s Resolution rule.

**Execution**: construct all possible connected & maximal tilings then evaluate them.
Encoding proof-structures

Computational content of proofs

\[ \varphi \otimes \]
Encoding proof-structures

*Computational content of proofs*
Encoding proof-structures

Computational content of proofs

\[ \text{ax} \quad \text{ax} \quad \text{ax} \]
\[ 1 \quad 2 \quad 3 \quad 4 \quad 5 \quad 6 \]
\[ \Rightarrow \quad \otimes \quad \Rightarrow \]
\[ 7 \quad 8 \]

\[ \text{cut} \quad \text{cut} \]

\[ R \quad \star \]
\[ \mathcal{I} \quad \mathcal{L} \quad \mathcal{I} \quad \mathcal{L} \]

Ex
Encoding proof-structures

Computational content of proofs
Encoding proof-structures

Logical content of proofs

\[ \mathcal{L} \otimes \text{cut} \]

\[ \mathcal{R} \otimes \text{cut} \]

Danos-Regnier correctness: is axioms+test a tree for any test?

Stellar logical correctness: does \( \text{Ex}(\text{uniEF26} \text{ax} S \top \text{uniEF26} \text{test} S, \phi) \) satisfy some property \( P \)?

\[ \text{↰ MLL :} | \text{Ex}(\text{uniEF26} \text{ax} S \top \text{uniEF26} \text{test} S, \phi) | = \text{one.pnum.} \]

\[ \text{↰ MLL+MIX :} \text{Ex}(\text{uniEF26} \text{ax} S \top \text{uniEF26} \text{test} S, \phi) \text{ terminates.} \]

Orthogonality. \( \text{Ex}(\text{uniEF26} \text{one.pnum} \top \text{uniEF26} \text{two.pnum}) \) satisfies \( P \) \( \iff \) \( \text{uniEF26} \text{one.pnum} \bot \text{uniEF26} \text{two.pnum} \).
Encodings proof-structures

Logical content of proofs

Danos-Regnier correctness: is axioms+test a tree for any test?
Encoding proof-structures

Logical content of proofs

Danos-Regnier correctness: is axioms+test a tree for any test?
Stellar logical correctness: does \( \text{Ex}(\Phi_{\text{ax}} \cup \Phi_{\text{test}}) \) satisfy some property \( P \)?
Encoding proof-structures

Logical content of proofs

Danos-Regnier correctness: is axioms+test a tree for any test?

Stellar logical correctness: does $\text{Ex}(\Phi_{\text{ax}} \cup \Phi_{\text{test}}_{\mathcal{I},\varphi})$ satisfy some property $P$?

$\Downarrow$ MLL: $|\text{Ex}(\Phi_{\text{ax}}_{\mathcal{I}} \cup \Phi_{\text{test}}_{\mathcal{I},\varphi})| = 1$. 
Encoding proof-structures

*Logical content of proofs*

Danos-Regnier correctness: is axioms+test a tree for any test?

Stellar logical correctness: does $\text{Ex}(\Phi^\text{ax}_\mathcal{I} \cup \Phi^\text{test}_\mathcal{I},\phi)$ satisfy some property $P$?

- $\text{MLL}: |\text{Ex}(\Phi^\text{ax}_\mathcal{I} \cup \Phi^\text{test}_\mathcal{I},\phi)| = 1$.
- $\text{MLL+MIX}: \text{Ex}(\Phi^\text{ax}_\mathcal{I} \Phi^\text{test}_\mathcal{I},\phi)$ terminates.
Encoding proof-structures

Logical content of proofs

\[ \begin{array}{c}
\text{ax} & 1 \\
\text{x} & 2 \\
\text{ax} & 3 \\
\text{ax} & 4 \\
\text{ax} & 5 \\
\text{ax} & 6 \\
\downarrow \text{cut} & \\
\downarrow \text{cut} & \\
7 & \text{cut} \\
8 & \text{cut} \\
\end{array} \]

Danos-Regnier correctness: is axioms+test a tree for any test?

Stellar logical correctness: does \( \text{Ex}(\Phi^\text{ax} \cup \Phi^\text{test}, \varphi) \) satisfy some property \( P \)?

\( \text{MLL:} \quad |\text{Ex}(\Phi^\text{ax} \cup \Phi^\text{test}, \varphi)| = 1. \)

\( \text{MLL+MIX:} \quad \text{Ex}(\Phi^\text{ax} \cup \Phi^\text{test}, \varphi) \) terminates.

Orthogonality. \( \text{Ex}(\Phi_1 \cup \Phi_2) \) satisfies \( P \iff \Phi_1 \perp \Phi_2. \)
Two notions of type

*Unified in the same framework*

Types as labels (type theory). $A, B ::= X_i | X_i^\perp | A \otimes B | A \between B$. 

Infininitely many (sub)types + ∈ $A$ usually undecidable vs #: $A$ usually decidable.

Related by adequacy: $\text{Tests}(A) \subseteq A$. 

Tests $(A)$
Two notions of type

*Unified in the same framework*

Types as labels (type theory). \( A, B ::= X_i \mid X_i^\perp \mid A \otimes B \mid A \bowtie B. \)

\[ \downarrow \quad A \mapsto \text{Tests}(A) \text{ finite} \quad \Phi \text{ logically correct } \iff \Phi \perp \text{Tests}(A). \]
Two notions of type

*Unified in the same framework*

Types as labels (type theory). $A, B ::= X_i \mid X_i \perp \mid A \otimes B \mid A \bowtie B$.

$\Downarrow A \mapsto \text{Tests}(A)$ finite  $\Phi$ logically correct $\iff \Phi \perp \text{Tests}(A)$.

Types as behaviour classes (realisability).
Two notions of type

Unified in the same framework

Types as labels (type theory). \( A, B ::= X_i \mid X_i^\perp \mid A \otimes B \mid A \# B. \)

\( \downarrow A \mapsto \text{Tests}(A) \) finite \( \Phi \) logically correct \( \iff \Phi \perp \text{Tests}(A). \)

Types as behaviour classes (realisability).

- Pre-type: set of constellation \( A; \)
Two notions of type

*Unified in the same framework*

**Types as labels (type theory).** $A, B ::= X_i \mid X_i^\perp \mid A \otimes B \mid A \bowtie B$.

$\Downarrow A \leftrightarrow \text{Tests}(A)$ finite

$\Phi$ logically correct $\iff \Phi \perp \text{Tests}(A)$.

**Types as behaviour classes (realisability).**

- Pre-type: set of constellation $A$;
- Orthogonal: $A^\perp$ (dual constellations);
Two notions of type

*Unified in the same framework*

**Types as labels (type theory).** $A, B ::= X_i \mid X_i^\perp \mid A \otimes B \mid A \multimap B.$

$\Downarrow$ $A \mapsto \text{Tests}(A)$ finite  \hspace{1cm} $\Phi$ logically correct $\iff$ $\Phi \perp \text{Tests}(A)$.

**Types as behaviour classes (realisability).**

- Pre-type: set of constellation $\text{A}$;
- Orthogonal: $\text{A}^\perp$ (dual constellations); Conduct: $\text{A} = \text{A}^\perp^\perp$;
Two notions of type

Unified in the same framework

Types as labels (type theory). \( A, B ::= X_i \mid X_i^\bot \mid A \otimes B \mid A \Rightarrow B. \)
\( \vdash A \leftrightarrow \text{Tests}(A) \) finite \( \Phi \) logically correct \( \iff \Phi \bot \text{Tests}(A). \)

Types as behaviour classes (realisability).

- Pre-type : set of constellation \( A \);
- Orthogonal : \( A^\bot \) (dual constellations); Conduct : \( A = A^{\bot\bot} \);
- Tensor : \( A \otimes B = \{ \Phi_A \cup \Phi_B, \Phi_A \in A, \Phi_B \in B \}^{\bot\bot}. \)

\( \Downarrow \) \( A \) \( \text{Tests}(A) \) \( \bot \) \( \subseteq A \).
Two notions of type

*Unified in the same framework*

**Types as labels (type theory).** $A, B ::= X_i | X_i^\bot | A \otimes B | A \triangleleft B.$

\[ A \leftrightarrow \text{Tests}(A) \text{ finite} \quad \Phi \text{ logically correct} \iff \Phi \perp \text{Tests}(A). \]

**Types as behaviour classes (realisability).**

- Pre-type: set of constellation $A$;
- Orthogonal: $A^\bot$ (dual constellations); Conduct: $A = A^{\bot\bot}$;
- Tensor: $A \otimes B = \{ \Phi_A \cup \Phi_B, \Phi_A \in A, \Phi_B \in B \}^{\bot\bot}$.

Infinitely many (sub)types + $\Phi \in A$ usually undecidable vs $\Phi : A$ usually decidable.
Two notions of type

*Unified in the same framework*

**Types as labels (type theory).** \( A, B ::= X_i \mid X_i^\perp \mid A \otimes B \mid A \Rightarrow B. \)
\( \Downarrow \) \( A \mapsto \text{Tests}(A) \) finite \( \Phi \) logically correct \iff \( \Phi \perp \text{Tests}(A). \)

**Types as behaviour classes (realisability).**

- Pre-type : set of constellation \( A \);
- Orthogonal : \( A^\perp \) (dual constellations); Conduct : \( A = A^\perp\perp \);
- Tensor : \( A \otimes B = \{ \Phi_A \cup \Phi_B, \Phi_A \in A, \Phi_B \in B \}^\perp\perp. \)

Infinitely many (sub)types + \( \Phi \in A \) usually undecidable *vs* \( \Phi : A \) usually decidable.
Related by *adequacy* : \( \text{Tests}(A)^\perp \subseteq A. \)
Technical development

Current works / In progress.
Technical developement

Current works / In progress.

• formal definition of stellar resolution & properties;
Technical development

Current works / In progress.

- formal definition of stellar resolution & properties;
- encoding of several models (automata, circuits, tiling models, ...);
Technical development

Current works / In progress.

- formal definition of stellar resolution & properties;
- encoding of several models (automata, circuits, tiling models, ...);
- model of MLL(+MIX) and IMELL (Intuitionistic exponentials);
Technical development

Current works / In progress.

- formal definition of stellar resolution & properties;
- encoding of several models (automata, circuits, tiling models, ...);
- model of MLL(+MIX) and IMELL (Intuitionistic exponentials);

Future works.
Technical development

Current works / In progress.

• formal definition of stellar resolution & properties;
• encoding of several models (automata, circuits, tiling models, ...);
• model of MLL(+MIX) and IMELL (Intuitionistic exponentials);

Future works.

• New point of view for first/second order logic + additives + neutrals;
Technical development

Current works / In progress.

- formal definition of stellar resolution & properties;
- encoding of several models (automata, circuits, tiling models, ...);
- model of MLL(+MIX) and IMELL (Intuitionistic exponentials);

Future works.

- New point of view for first/second order logic + additives + neutrals;
- Implicit computational complexity analysis.
Technical development

Current works / In progress.
  • formal definition of stellar resolution & properties;
  • encoding of several models (automata, circuits, tiling models, ...);
  • model of MLL(+MIX) and IMELL (Intuitionistic exponentials);

Future works.
  • New point of view for first/second order logic + additives + neutrals;
  • Implicit computational complexity analysis.

Thank you for listening.