Linear Logic and Strong Normalization

Beniamino Accattoli

Carnegie Mellon University
History

- **Girard, TCS ’87**: linear logic (LL) and strong normalization (SN).
- A crucial lemma about the exponentials was left **unproven**.
- **Danos, PhD ’90**: elaborated proof for second order MELL.
- Various other people worked on **SN for LL**:
  - Joinet, van Raamsdonk, Okada, Di Cosmo & Guerrini.
- **Tortora de Falco and Pagani, TCS ’10**: SN for second order LL.
- **Complex and long proof**, requiring confluence.
- **Here**: a simple and understandable proof, no need for confluence.
1. Strong normalization, commutative cases, and proof nets

2. Proof nets and substitution

3. The axiomatic proof

4. New presentation of proof nets
Kinds of cut

There are two kinds of cut-elimination cases.

1) **Principal**, i.e. the last rules introduce the cut formulas:

\[
\begin{align*}
\pi & : \vdash \Gamma, A \\
\theta & : \vdash \Delta, A^\perp \\
\Gamma, \Delta & \vdash \vdash \text{cut}
\end{align*}
\]

\[
\begin{align*}
\pi & : \vdash ?\Gamma, A \\
\theta & : \vdash \Delta, A^\perp \\
\Gamma, \Delta & \vdash \vdash \text{cut}
\end{align*}
\]

2) **Commutative**, one last rule has no relation with the cut formula:

\[
\begin{align*}
\pi & : \vdash \Gamma, A \\
\theta & : \vdash \Delta, B \\
\Gamma, \Delta & \vdash \vdash \text{cut}
\end{align*}
\]

\[
\begin{align*}
\pi & : \vdash ?\Gamma, A \\
\theta & : \vdash \Delta, A^\perp \\
\Gamma, \Delta & \vdash \vdash \text{cut}
\end{align*}
\]

\[
\begin{align*}
\pi & : \vdash ?\Gamma, A \\
\theta & : \vdash ?\Delta, B \\
\Gamma, \Delta & \vdash \vdash \text{cut}
\end{align*}
\]
Commutative cases

- Commutative cases are **the burden** of cut-elimination.
- **Problem**: the cut rule commutes with itself.
- **Consequence**: silly diverging reductions.
- **Solution**: Switch to proof nets, where commutative cases (mostly) disappear.
The **multiplicative fragment**:

- From sequent calculus to proof nets

- **ax**

- **cut**

- **mix**

- **⊗**

- **⇝**

- **θ**

- **π**

- **Δ**

- **Γ**

- **⊥**
The exponential fragment:

\[
\begin{align*}
\pi : \vdash \Gamma & \quad \xrightarrow{w} \quad \vdash \Gamma, \Diamond A \\
\pi : \vdash \Gamma & \quad \xrightarrow{\pi^*} \quad \pi : \vdash \Gamma, A \\
\pi : \vdash \Diamond \Gamma, A & \quad \xrightarrow{!} \quad \vdash \Diamond \Gamma, !A
\end{align*}
\]
Girard introduced boxes according to the **black-box principle**:

"boxes are treated in a perfectly modular way: we can use the box B without knowing its contents, i.e., another box B' with exactly the same doors would do as well"

**Principal cases**: 2 deductive rules cut at level 0 in the same box.

**Only one commutative case**: a rule moving boxes to bring premises of a cut at the same box level
Proof nets cut-elimination: principal cases

\[
\begin{align*}
\text{ax} & \quad A \quad A \quad \text{cut} \quad A \\
\text{cut} & \quad A \quad B \quad \text{cut} \quad A \\
\text{cut} & \quad W \quad ! \quad ?A \quad \text{cut} \quad ?B_1 \quad ?B_k \\
\text{cut} & \quad ?A \quad ?A \quad \text{cut} \quad ?B_1 \quad ?B_k \\
\text{cut} & \quad A \quad ! \quad \Gamma \\
\text{cut} & \quad A \quad ! \quad \Gamma \\
\end{align*}
\]
Girard’s original presentation of proof nets has a commutative case:

This rule is the source of all technical complications.
Outline

1. Strong normalization, commutative cases, and proof nets
2. Proof nets and substitution
3. The axiomatic proof
4. New presentation of proof nets
Exponentials and explicit substitutions

- **Statically:**
  In linear logic $A \Rightarrow B$ decomposes as $!A \multimap B$.

- **Dynamically:**
  $\beta$ splits in a **multiplicative cut** followed by an **exponential cut**.

- **Intuition:** exponentials = **explicit substitutions**.

- **Ordinary substitution** or **implicit substitution**: $t\{x/s\}$.

- **Explicit substitution**: $t[x/s]$.

- **Then:**

  $$(\lambda x . t)s \rightarrow^\beta t\{x/s\}$$

  becomes

  $$(\lambda x . t)s \rightarrow^m t[x/s] \rightarrow^e t\{x/s\}$$
What is a variable?

- $[x/s]$ is a $!$-box containing $s$.
- $t[x/s]$ is a cut between $t$ and the $!$-box around $s$.

**What is a variable?** a maximal tree of $?$-rules (crossing boxes).

**Example of explicit substitution** $t[x/s]$:

![Diagram showing explicit substitution](image)

- **Next slide**: definition of substitution in proof nets.
Example of substitution
Outline

1. Strong normalization, commutative cases, and proof nets
2. Proof nets and substitution
3. The axiomatic proof
4. New presentation of proof nets
Proof technique: reducibility candidates in bi-orthogonal form (Girard ’87).

The proof is axiomatic: it works for every set of rewriting rules satisfying the axioms.

For Girard’s rules the axioms are hard to prove.

I will later give a new set of rules for which the axioms are easy.
The axiomatic proof

The proof depends on 3 abstract properties of the rewriting relation $\rightarrow$:

1. **Substitution and promotion commute:**

   $$!(P\{x/Q\}) \rightarrow^* (!P)\{x/Q\}$$

2. **Full composition:**

   $$P[x/Q] \rightarrow^+ P\{x/Q\}$$

3. **Kesner’s IE property:**

   $$\frac{P\{x/Q\} \in SN \quad Q \in SN \quad P[x/Q] \in SN}{P[x/Q] \in SN}$$

These properties hold in the **untyped case**.
The IE property

- Key property of \( \lambda \)-calculus:

\[
\frac{t\{x/s\}u_1 \ldots u_n \in SN_\beta \quad s \in SN_\beta}{(\lambda x.t)su_1 \ldots u_n \in SN_\beta}
\]

called the fundamental lemma of perpetuality by van Raamsdonk, Severi, Sorensen, and Xi.

- It is more or less explicitly used in all proofs of SN, e.g. van Daalen’s for simple types, or Girard’s for system F.

- Key point in inductive definitions of the set of SN \( \lambda \)-terms (van Raamsdonk & Severi, Loader).

- Kesner, LMCS ’09:
  Preservation of SN for exp. subst. reduces to the IE property:

\[
\frac{t\{x/s\}u_1 \ldots u_n \in SN_\beta \quad s \in SN_\beta}{t[x/s]u_1 \ldots u_n \in SN_\beta}
\]
Key point of the new proof

The proof is by **induction on the structure** of the net.

The difficult case is for **promotion**.

**Inductive Hypothesis**: \( !(P[x/Q]) \in SN \rightarrow (\text{and } Q \in SN \rightarrow) \).

**Goal**: \( (!P)[x/Q] \in SN \rightarrow \).

**Key point of the proof**:

\[
\begin{align*}
(P[x/Q]) & \rightarrow^+ !(P\{x/Q\}) \in SN & \text{by full composition and i.h.} \\
& \rightarrow^* (!P\{x/Q\}) \in SN & \text{by commutation} \\
& \text{implies } (!P)[x/Q] \in SN & \text{by the IE property}
\end{align*}
\]

**Novelty**: no analysis of the reducts of \( !(P[x/Q]) \).
Main difficulty for the additives: they are not confluent.

All previous proofs of SN use confluence.

That’s why T. de Falco and Pagani’s proof is very technical.

Here: the first proof of SN not requiring confluence.

Consequence: it smoothly scales up to the additives.
Outline

1. Strong normalization, commutative cases, and proof nets
2. Proof nets and substitution
3. The axiomatic proof
4. New presentation of proof nets
Girard introduced boxes according to the **black-box principle**.

The black-box principle induces a **commutative case**.

In such a case **the IE property is hard to prove**.

**No black-box** in the new approach.

**Consequences:**

1. Cuts can be reduced also when they **cross box borders**.
2. **No commutative case**.
3. **Easy proof of the IE property**.
Box-crossing rules 1

The rules act through possibly many box borders.
These two cases absorb the commutative case.
Proving the IE property

The proof of the IE property:

1. box-crossing rules: two lemmas, a simple induction on a triple.

2. black-box rules: many lemmas and pages, very technical.

Recall the possible interactions with a graphical variable/?-tree:
Comparing inductive cases

**Black-box rules:**

\[ \text{Box-crossing rule:} \]

**Intuition:** the commutative rule breaks the explicit substitution form.
Commutation of promotion and substitution

The property:

\[(P\{x/Q\}) \rightarrow^* (!P)\{x/Q\}\]

It follows immediately from the addition of the following rules:

\[\sim_{pc} \quad \sim_{cw} \quad \rightarrow_{pw} \quad w\]

That are *semantically sound* and *needed* to represent λ-terms.
Further extension

- In the paper I also consider the following **optional** rules:

  ![Diagram](diagram.png)

  - **not present** in Tortora de Falco and Pagani’s proof.
  - Usually, their addition requires **delicate and sophisticated reasoning** (Di Cosmo & Guerrini, Tranquilli & Pagani).
  - Here it is almost **transparent**.
Ideas

- **Kesner, LMCS '09:**
  IE technique for SN of explicit substitutions.

- **A.-Guerrini, CSL '09:**
  box-free PN for $\lambda$-terms with explicit substitutions.

- **A.-Kesner, CSL '10:**
  1. new approach to explicit substitutions (structural $\lambda$-calculus $\lambda_j$).
  2. IE technique applies extremely easily to $\lambda_j$.

- **Here, RTA '13:**
  back to PN, generalizing Kesner's technique and its application.
Conclusions

Summing up:

1. A neat understanding of substitution for proof nets (PN).
2. A simple axiomatic proof of strong normalization for LL.
3. A new presentation of PN s.t. the axioms are easy to verify.
5. A fruitful interaction between LL and explicit substitutions.
THANKS!