Differential Nets: a Paradigm of Non-Determinism?

Paolo Tranquilli

Dipartimento di Matematica
Università degli Studi Roma Tre

Preuves, Programmes et Systèmes
Université Denis Diderot - Paris 7

Ecole Jeunes Chercheurs en Informatique Mathématique

/2008

Outline

Outline

1. **Linear? Differential LL does it better**
   - Why Linear Logic is linear?
   - Differential Linear Logic
2. **Nets and Calculi by examples**
   - Proof nets and Lambda-Calculus
   - Differential Nets and Resource Calculus
   - Differential Nets and π-Calculus
3. **Conclusion**

Vector spaces and LL: is there a link?

When one hears a linear logician talk, one notes an extensive use of jargon borrowed from vector spaces and analysis:

Linear, dual ($A^{\perp}$), exponential, tensor ($\otimes$), direct sum ($\oplus$)

What is the catch?

The link is there, though somewhat vague

- The ground for LL (Girard, 1987) were coherent spaces: Girard imagined formulas A as spaces given by atomic bits of information $|A|$
- Moreover, he imagined these bits to be a sort of basis for a vector space

Linearity

in Logic

using a hypothesis exactly once (no structural rules)

Curry Howard ⟷

in Computer Science

using an input/resource exactly once (no erasing/duplication)

in Analysis

$F(aU + bV) = aF(U) + bF(V)$ (with continuity)

...though the link with the third may seem a little vague...

Linearity

in Logic

using a hypothesis exactly once
(no structural rules)

Curry Howard

in Computer Science

using an input/resource exactly once (no erasing/duplication)

in Analysis

$F(aU + bV) =$
$aF(U) + bF(V)$
(with continuity)

...though the link with the third may seem a little vague...

Duality

- In vector spaces the base of duality is the scalar product $\langle \cdot , \cdot \rangle$
- In logic, it can be seen as the core interaction of the cut rule
- Coherence spaces are generated by $\langle U, V \rangle = \# U \cap V$, and $\{0, 1\}$ as polar set
- In finiteness spaces (Ehrhard 2003) the condition is relaxed: $\# U \cap V < \omega$

More linear than LL

- In fact, in vector spaces $\langle U, V\rangle = \sum_{x\in\mathcal{B}} U_x V_x$
- Easiest way to have finite sums: $\#\operatorname{supp} U \cap \operatorname{supp} V < \omega$. Take it as condition on supports
- R semiring with unit, $R\langle A\rangle$ are R-modules where all logic connectives really correspond to the vector space constructs
- For example: $R\langle A \multimap B\rangle \cong R\langle A\rangle \multimap R\langle B\rangle$, linear continuous morphisms (in linear topology, defined independently of the structure of finiteness spaces)

Linearity revisited

in Logic

Linear hypotheses, to be used exactly once

in Computer Science

Linear inputs, to be used exactly once

in Analysis

Linear arguments occurring in linear position

Linearity revisited

in Logic

Non-linear hypotheses, that can be contracted or weakened

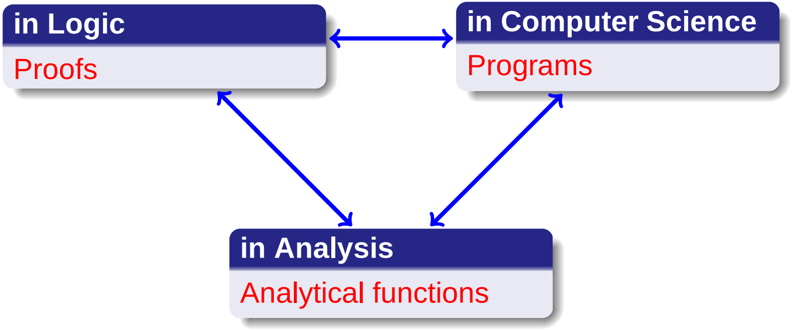
in Computer Science

Non-linear inputs, that can be duplicated or erased

in Analysis

Non-linear arguments occurring as powers

Linearity revisited



Formal sums

- Coherent spaces: $\#U \cap V \leq 1 \implies$ reducing a proof/program gives one proof/program (determinism)
- Finiteness spaces: result of an interaction is a generic (though finite) sum, i.e.

a proof/program can reduce to a sum of proofs/programs!

- What does it mean?

Interpretation of formal sums

- With $R = \mathbb{N}$:

Non-deterministic choice

$$\sigma \begin{array}{l} \nearrow \pi_1 \\ \searrow \pi_2 \end{array} \rightsquigarrow \sigma \to \pi_1 + \pi_2$$

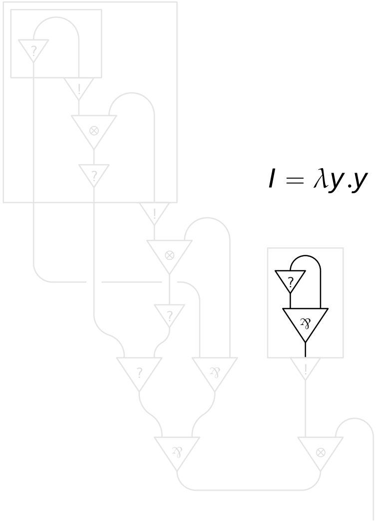
- Other interpretations are possible. With $R = \mathbb{R}^+$:

Probabilistic choice

$$\sigma \begin{array}{l} \overset{p}{\nearrow} \pi_1 \\ \underset{1-p}{\searrow} \pi_2 \end{array} \rightsquigarrow \sigma \to p\pi_1 + (1-p)\pi_2$$

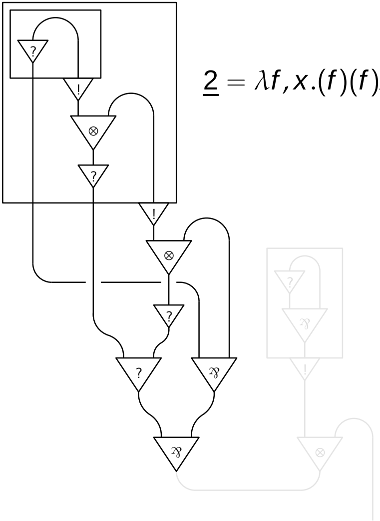
Outline

An example



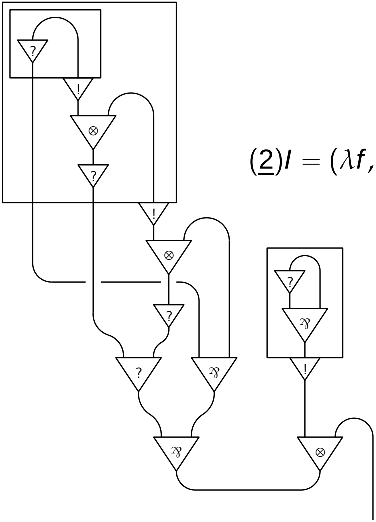
$$I = \lambda y.y$$

An example



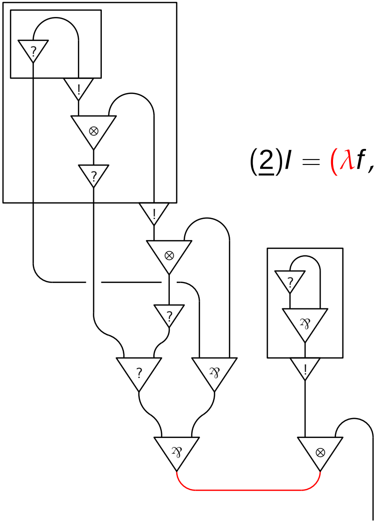
$$\underline{2} = \lambda f, x.(f)(f)x$$

An example



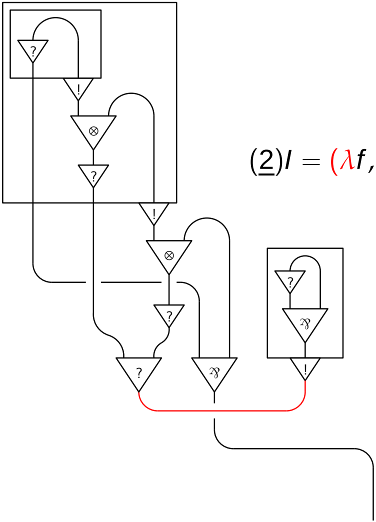
$$(\underline{2})I = (\lambda f, x.(f)(f)x)\lambda y.y$$

An example



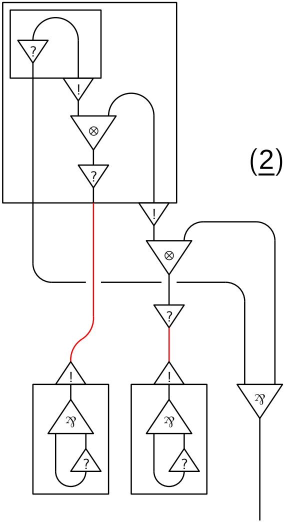
$$(\underline{2})I = (\lambda f, x.(f)(f)x)\lambda y.y$$

An example



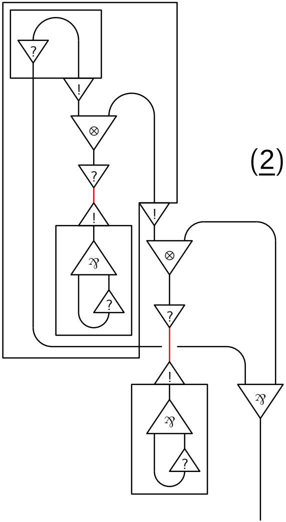
$$(\underline{2})I = (\lambda f, x.(f)(f)x)\lambda y.y$$

An example



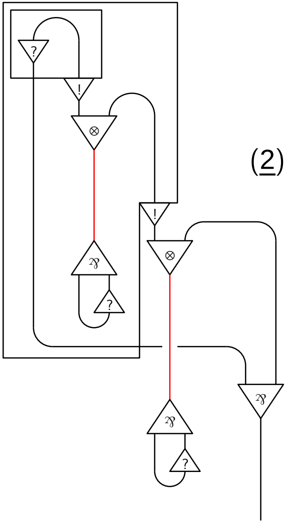
$$(\underline{2})I = (\lambda f, x.(f)(f)x)\lambda y.y$$

An example



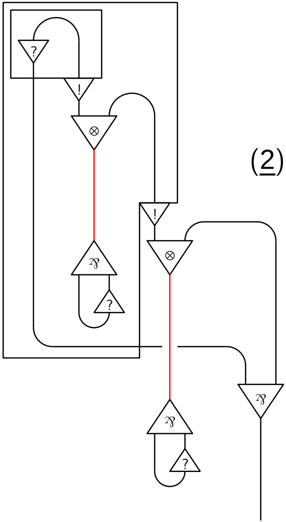
$$(\underline{2})I = ({\color{red}(\lambda} f,x.(f)(f)x)\lambda y.y$$

An example



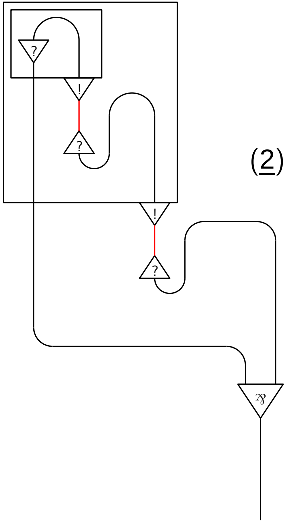
$$(\underline{2})I = (\lambda f, x.(f)(f)x)\lambda y.y$$

An example



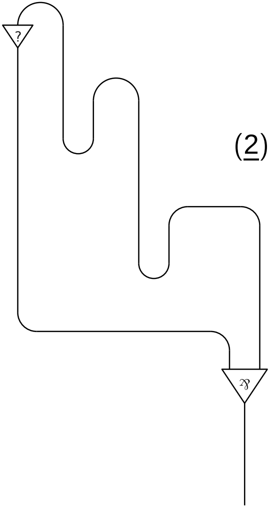
$$(\underline{2})I \xrightarrow{\beta} \lambda x.(\lambda y.y)(\lambda y.y)x$$

An example



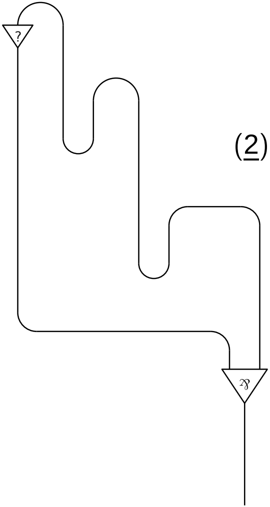
$$(\underline{2})I \xrightarrow{\beta} \lambda x.(\lambda y.y)(\lambda y.y)x$$

An example



$$(\underline{2})I \xrightarrow{\beta} \lambda x.(\lambda y.y)y.y)x$$

An example



$$(\underline{2})I \xrightarrow{\beta*} \lambda x.x$$

Inputs are packages

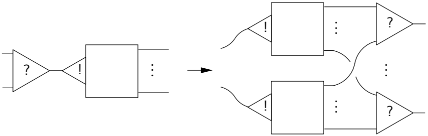
Inputs are packages that are

- duplicable
- erasable
- openable

Inputs are packages

Inputs are packages that are

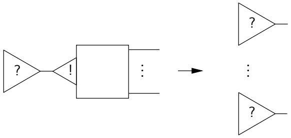
- duplicable
- erasable
- openable



Inputs are packages

Inputs are packages that are

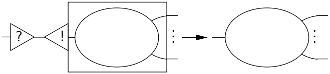
- duplicable
- erasable
- openable



Inputs are packages

Inputs are packages that are

- duplicable
- erasable
- openable



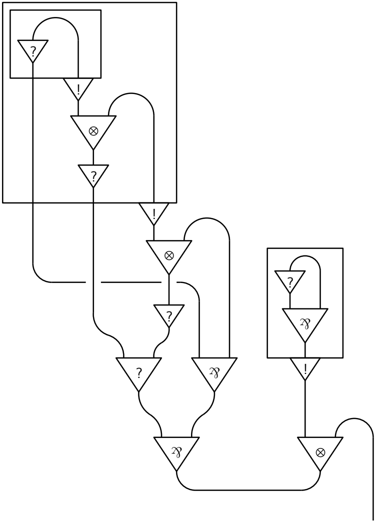
Single-use hypotheses/inputs

- What if a hypothesis/input is available just once (after which it “expires”), but is asked for more than once?
- We can imagine that it can be assigned non-deterministically (formal sums involved)
- We are able to see this in differential nets, the geometrical representation of Differential Linear Logic (Ehrhard and Regnier, 2004)

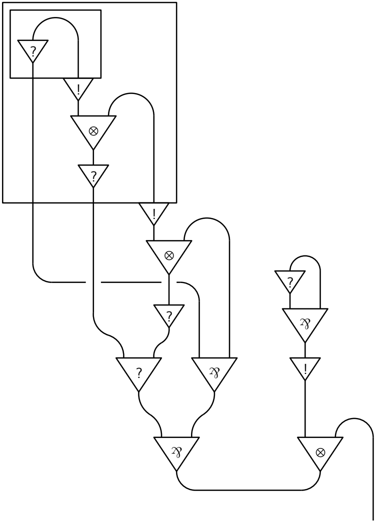
Single-use hypotheses/inputs

- What if a hypothesis/input is available just once (after which it "expires"), but is asked for more than once?
- We can imagine that it can be assigned non-deterministically (formal sums involved)
- We are able to see this in differential nets, the geometrical representation of Differential Linear Logic (Ehrhard and Regnier, 2004)

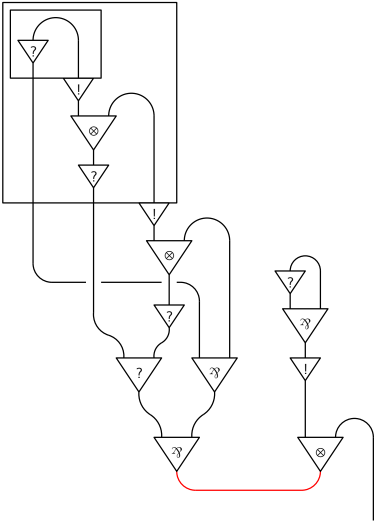
Back to an example



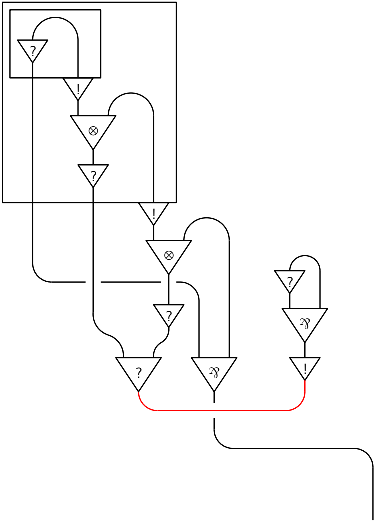
Back to an example



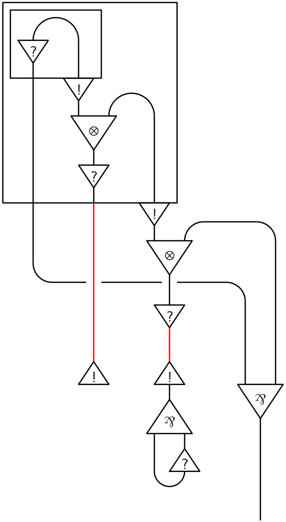
Back to an example



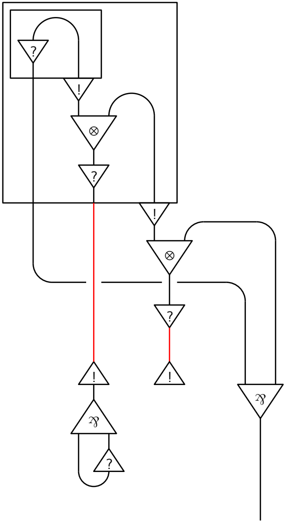
Back to an example



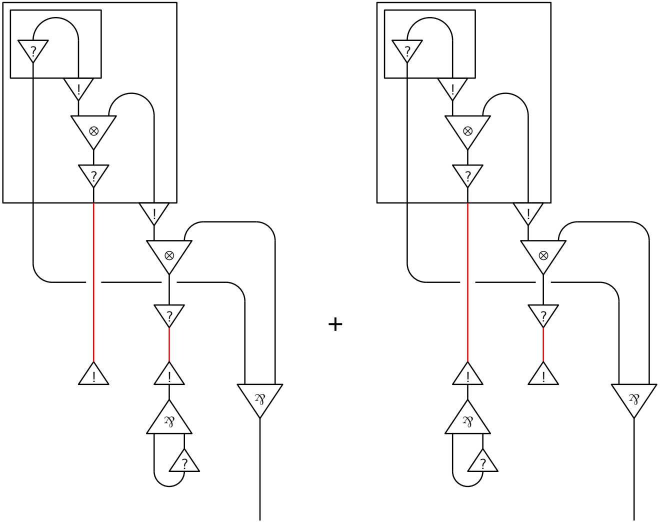
Back to an example



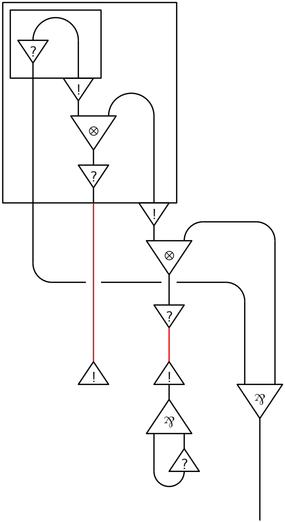
Back to an example



Back to an example

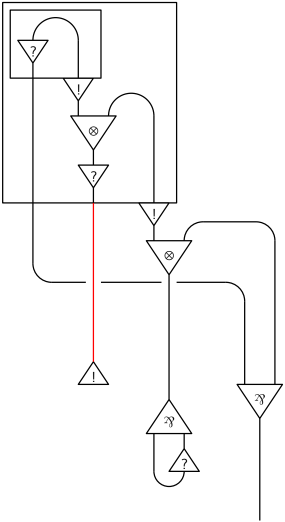


Back to an example

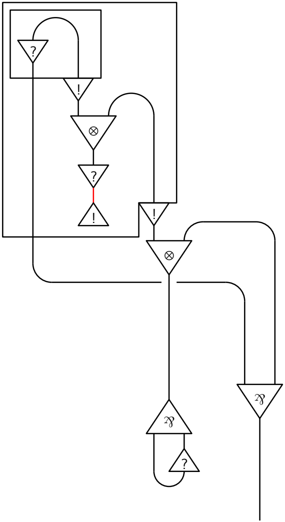


$+ \qquad 0$

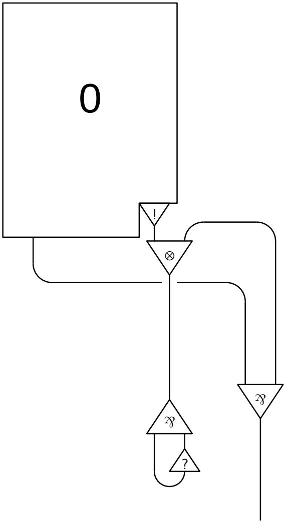
Back to an example



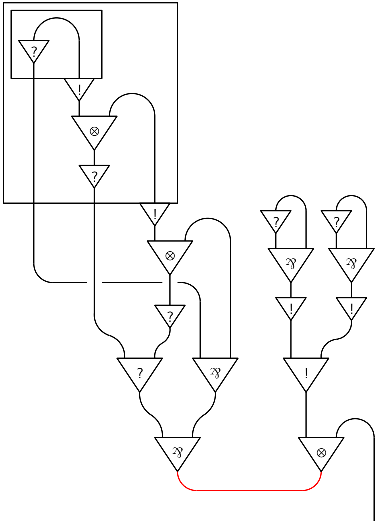
Back to an example



Back to an example

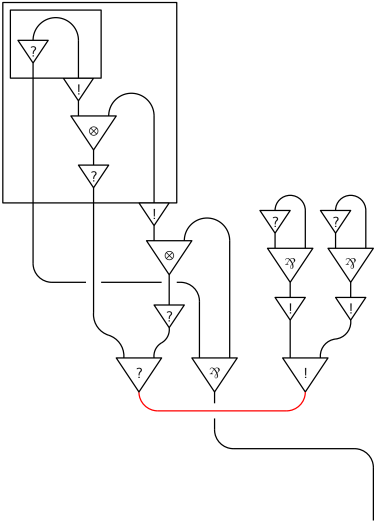


Yet another example

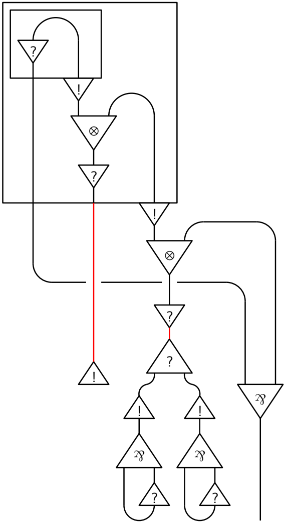


Differential Nets and Resource Calculus

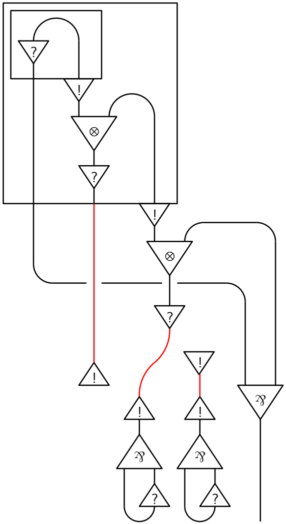
Yet another example



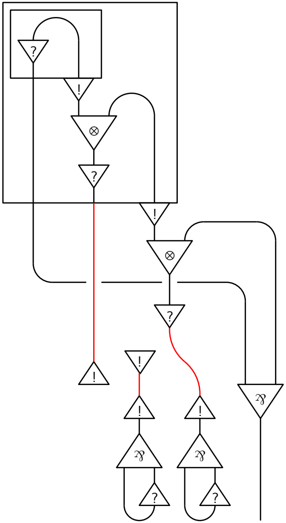
Yet another example



Yet another example



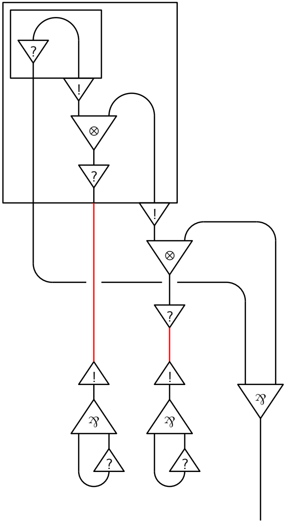
Yet another example



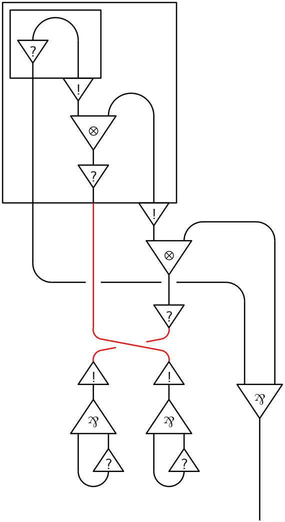
Yet another example

$$0$$

Yet another example

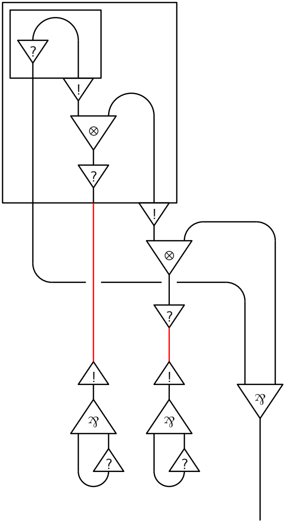


Yet another example



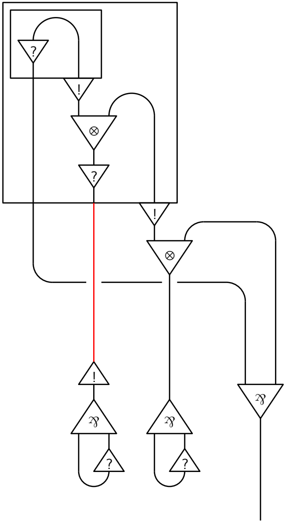
Yet another example

2.



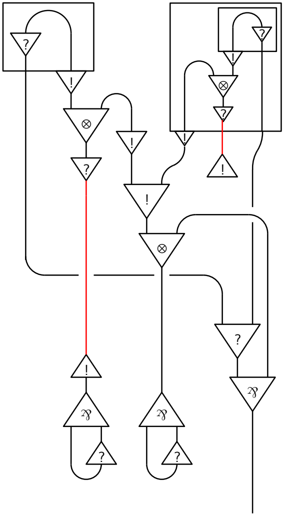
Yet another example

2·



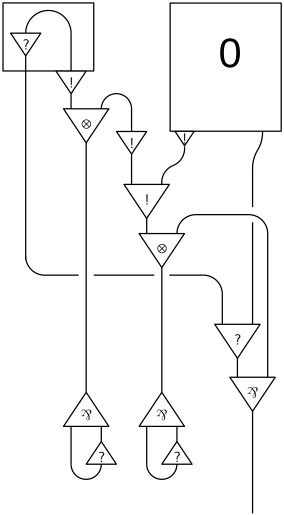
Yet another example

2.



Yet another example

2·



Resource Calculus

Does this behaviour correspond to a calculus like Proof Nets do for λ-calculus?

Yes, λ-calculus with multiplicities (Boudol, 1993), exactly presents this behaviour.

- In Boudol's calculus, arguments appear as *monomials* $u_1^{e_1} \dots u_k^{e_k}$ where $e_i \leq \infty$
- Example: $\langle \lambda f, x . \langle f \rangle (\langle f \rangle x^{\infty})^{\infty} \rangle (\lambda y . y)^2$
- Boudol's calculus naturally has a non-deterministic reduction. We use formal sums instead.

Resource Calculus

Does this behaviour correspond to a calculus like Proof Nets do for λ-calculus?

Yes, λ-calculus with multiplicities (Boudol, 1993), exactly presents this behaviour.

- In Boudol's calculus, arguments appear as *monomials* $u_1^{e_1} \ldots u_k^{e_k}$ where $e_i \leq \infty$
- Example: $\langle \lambda f, x. \langle f \rangle (\langle f \rangle x^\infty)^\infty \rangle (\lambda y.y)^2$
- Boudol's calculus naturally has a non-deterministic reduction. We use formal sums instead.

A sketch

- Single-use availability can also be thought as applying to the context of parallel processes
- When a process takes a communication channel, it is not available anymore to other ones
- Base of non-determinism in process calculi
- Intuition endorsed by translation of π-calculus into differential nets (Ehrhard and Laurent, 2007)

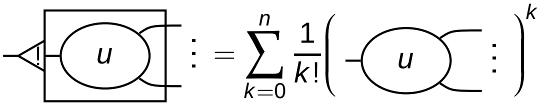
Outline

Cherry on the pie: some mathematical candy

- Linear substitution is differentiation:

$$\frac{\partial x^2 y}{\partial x} \cdot a = \frac{\partial x \cdot x \cdot y}{\partial x} \cdot a = a \cdot x \cdot y + x \cdot a \cdot y = 2xy \cdot a$$

(idea at the basis of Differential λ-Calculus (Ehrhard and Regnier, 2003))

- Exponential: 

(i.e. $u^\infty = \mathrm{e}^u$)

- Taylor expansion: $v = \sum_{t \in \mathcal{T}(v)} \frac{1}{\mathrm{m}(t)} t$ (Ehrhard, 2007)

Concluding remarks

- Differential linear logic and non-determinism seem tightly related, from its very beginning in the shift from coherent spaces to finiteness spaces
- On the other hand, non-determinism by formal sums may seem a little awkward from the computational point of view (for example one sees too many reductions to 0 that formal sums simply disregard)
- My (humble) opinion: developping a fully non-deterministic lazy reduction on differential nets that avoids “avoidable” reductions to 0 might help
- In any case: there are still many things to do!