Rewriting Aspects of Differential Linear Logic

Paolo Tranquilli

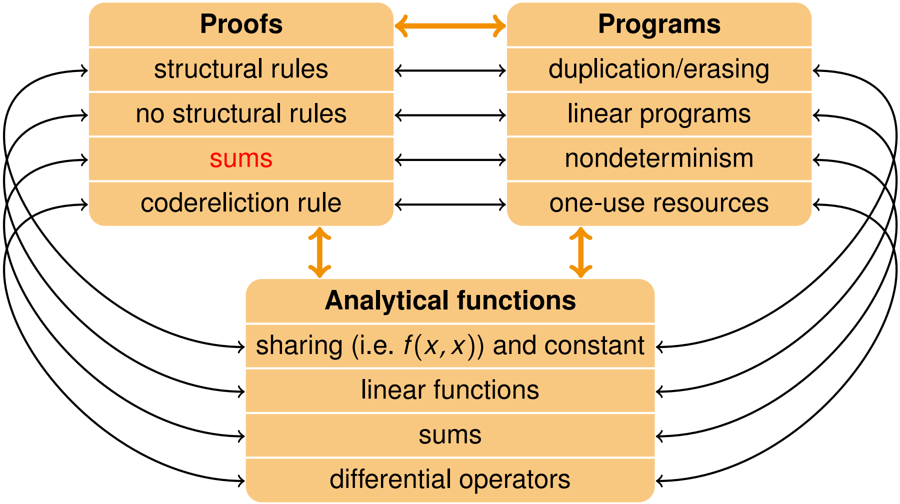
paolo.tranquilli@ens-lyon.fr

Laboratoire de l'Informatique du Parallélisme

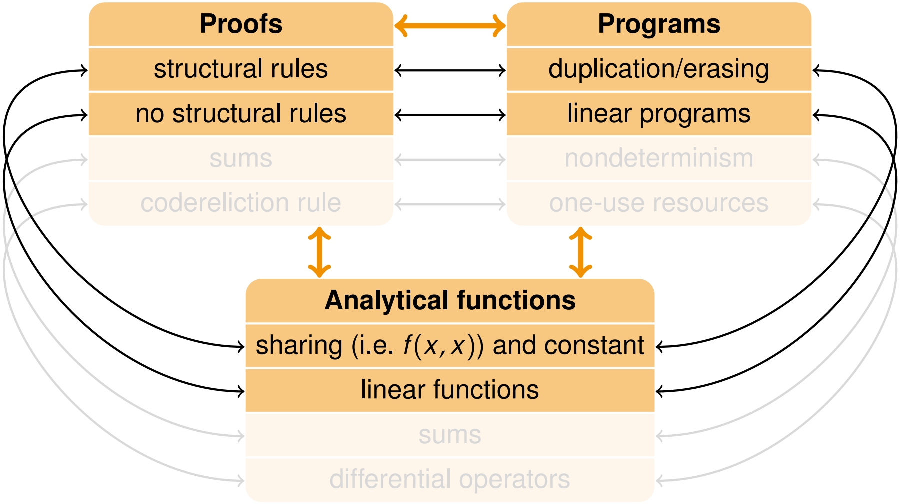
École Normale Supérieure de Lyon

FMCS, Kananaskis, Alberta, 11–14/06/2011

Proofs/programs as analytical functions



Ordinary proof nets



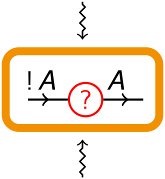
Proof nets as string diagrams

- Proof nets are a parallel syntax that abstracts away topologically commutation of rules, giving a neat account of cut-elimination.
- We can see them as string diagrams for monoidal categories with equality of morphisms $\leftrightsquigarrow$ topological equivalence.
- Linear category $\rightsquigarrow$
  monoidal comonad $! +$ isomorphism $!(A \mathbin{\&} B) \cong {!A} \otimes {!B} + \dots$
  $\rightsquigarrow$ further equalities of morphisms.
- Following will be a brief presentation of MELL proof nets with categorical and analysis intuitions.
- $A \to B \rightsquigarrow$ linear functions, $!A \to B \rightsquigarrow$ analytical functions.

Dereliction

!'s counit:

$$\epsilon : !A \to A$$



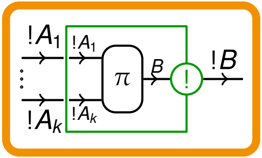
the identity x seen as analytical

Exponential box

!'s functoriality, multiplication (digging) and monoidalness:

$$\delta : !A \to !!A \qquad m : !A \otimes !B \to !(A \otimes B) \qquad \frac{\sigma : A \to B}{!\sigma : !A \to !B}$$

$$\pi^! : \bigotimes_i !A_i \xrightarrow{\otimes\delta} \bigotimes_i !!A_i \xrightarrow{m} !\left(\bigotimes_i !A_i\right) \xrightarrow{!\pi} !B$$



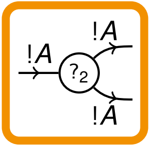
composition of analytical f's is analytical
box packages function to be ready for plugging

Structural rules

!'s functoriality, diagonal, terminal object, exponential isomorphism:

$\Delta : A \to A \,\&\, A$

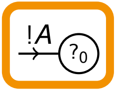
$d : {!A} \xrightarrow{!\Delta} {!(A \,\&\, A)} \cong {!A} \otimes {!A}$



$f(x, y)$ analyt. $\Longrightarrow$ $f(x, x)$ analyt.

$T : A \to \top$

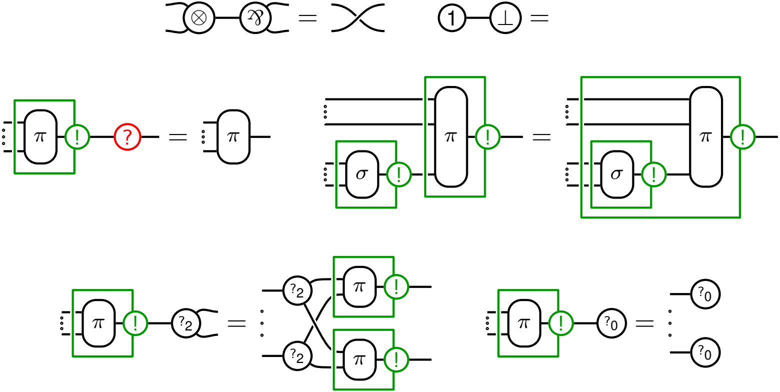
$e : {!A} \xrightarrow{!T} {!\top} \cong 1$



constant is analyt.

Contraction and weakening make $!A$ a commutative comonoid.

From equalities. . . to rewriting

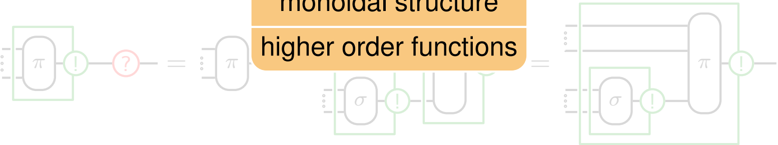


from string diagrams. . . to interaction nets!

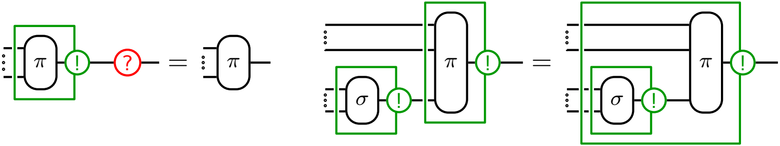
From equalities. . . to rewriting

monoidal structure

higher order functions

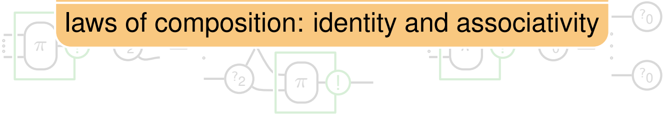


from string diagrams. . . to interaction nets!



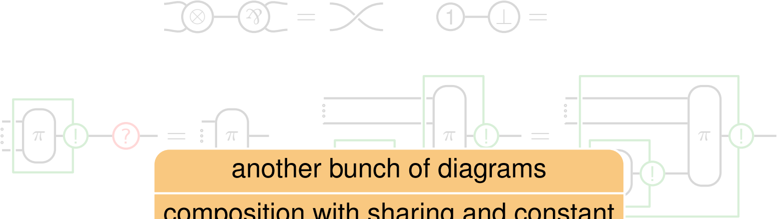
lots of diagrams

laws of composition: identity and associativity



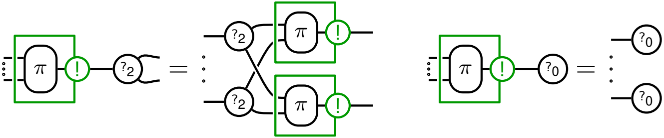
from string diagrams. . . to interaction nets!

From equalities. . . to rewriting



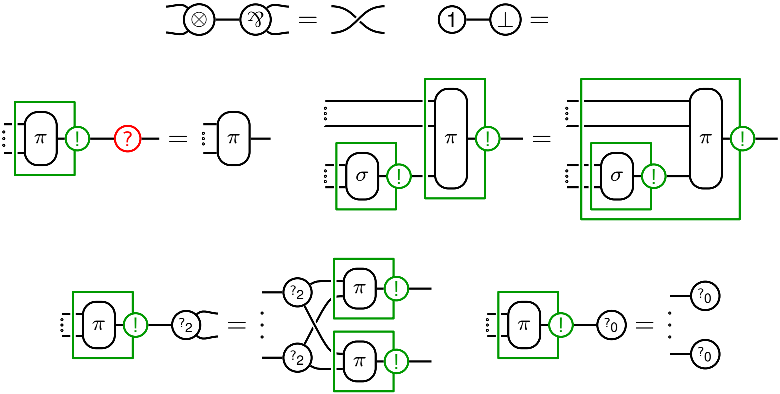
another bunch of diagrams

composition with sharing and constant



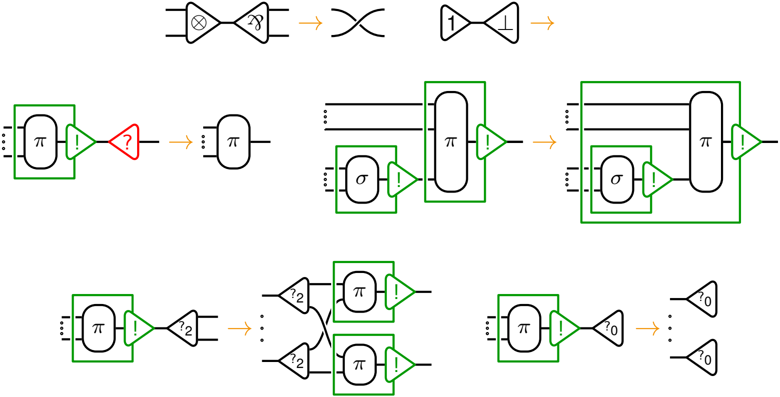
from string diagrams. . . to interaction nets!

From equalities. . . to rewriting



from string diagrams. . . to interaction nets!

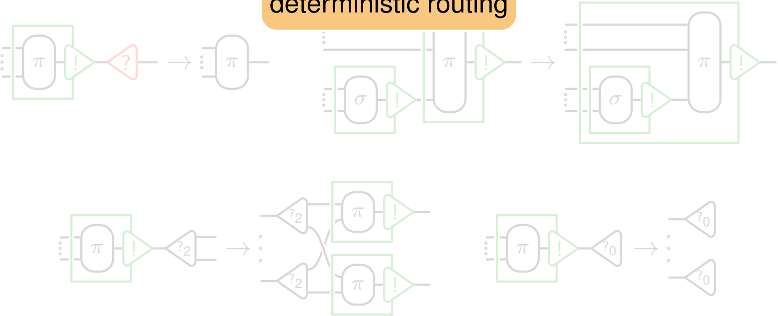
From equalities. . . to rewriting



from string diagrams. . . to interaction nets!

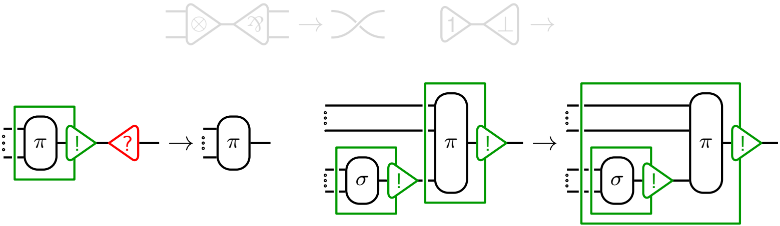
From equalities. . . to rewriting

deterministic routing



from string diagrams. . . to interaction nets!

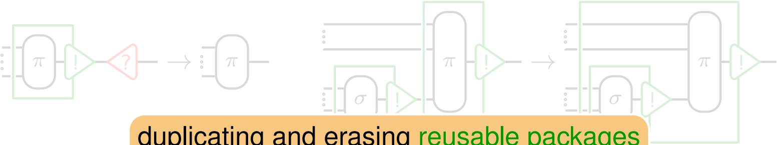
From equalities. . . to rewriting



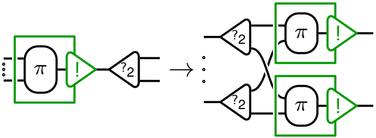
querying and fusing reusable packages

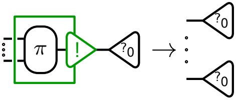
from string diagrams. . . to interaction nets!

From equalities. . . to rewriting



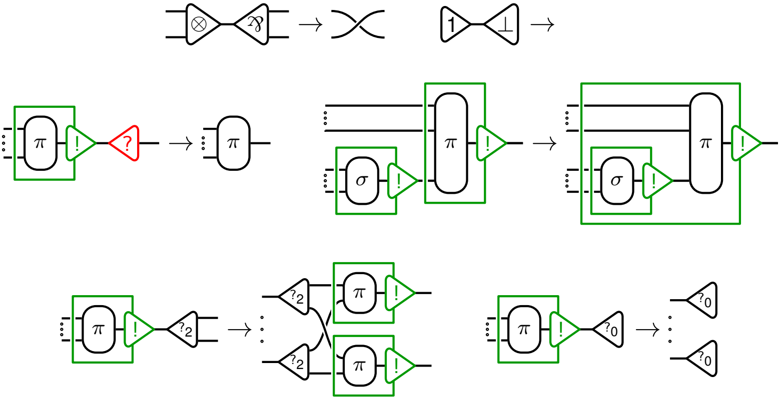
duplicating and erasing reusable packages





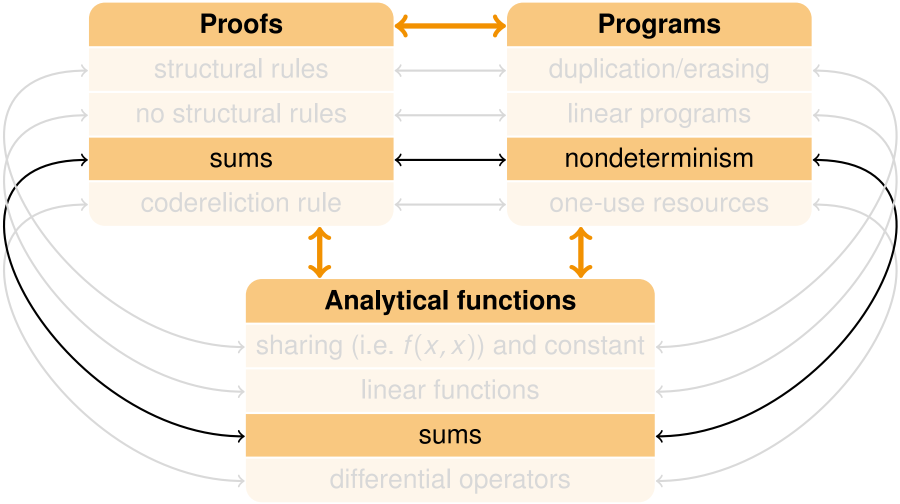
from string diagrams. . . to interaction nets!

From equalities. . . to rewriting



from string diagrams. . . to interaction nets!

Adding nondeterminism



Sums from biproducts

- What happens if product & and coproduct $\oplus$ are equal?
- We have the biproduct $\divideontimes$ (and $0 = \top$).
- Then morphisms $A \to B$ get a commutative monoid structure: in short, can be summed.

$$\pi + \sigma : A \xrightarrow{\Delta} A \divideontimes A \xrightarrow{\pi \divideontimes \sigma} B \divideontimes B \xrightarrow{\nabla} B, \qquad 0 : A \to \top \to B.$$

- Sum distributes on composition:

$$\left(\sum_i \pi_i\right); \left(\sum_j \sigma_j\right) = \sum_{ij}(\pi_i; \sigma_j),$$

as morphisms are linear!

Sums from biproducts

- What happens if product & and coproduct $\oplus$ are equal?
- We have the biproduct $\ast$ (and $0 = \top$).
- Then morphisms $A \to B$ get a commutative monoid structure: in short, can be summed.

$$\pi + \sigma : A \xrightarrow{\Delta} A \ast A \xrightarrow{\pi \ast \sigma} B \ast B \xrightarrow{\nabla} B, \qquad 0 : A \to \top \to B.$$

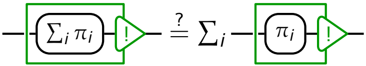
- Sum distributes on composition:

$$-\boxed{\textstyle\sum_i \pi_i}-\boxed{\textstyle\sum_j \sigma_j}- \;=\; \textstyle\sum_{ij} -\boxed{\pi_i}-\boxed{\sigma_j}-$$

as morphisms are linear!

Sums, nondeterminism, boxes

- Computationally, $\pi + \sigma$ can be viewed as internal choice between the two, or as independent parallel computations.
- What about boxes?

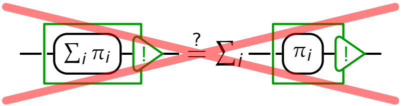


- Application of analytical functions/ordinary programs is linear in the function, but not in the argument:

$$(\lambda x.M)(N_1 + N_2) \neq (\lambda x.M)N_1 + (\lambda x.M)N_2.$$

Sums, nondeterminism, boxes

- Computationally, $\pi + \sigma$ can be viewed as internal choice between the two, or as independent parallel computations.
- What about boxes?



- Application of analytical functions/ordinary programs is linear in the function, but not in the argument:

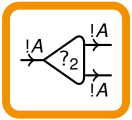
$$(\lambda x.M)(N_1 + N_2) \neq (\lambda x.M)N_1 + (\lambda x.M)N_2.$$

Costructural rules

!'s functoriality, codiagonal, coterminal object, exponential isomorphism:

$$\Delta : A \to A \ast A$$

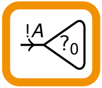
$$d : !A \xrightarrow{!\Delta} !(A \ast A) \cong !A \otimes !A$$



$f(x,y)$ analyt. $\Longrightarrow$ $f(x,x)$ analyt.

$$A \to \top$$

$$e : !A \to !\top \cong 1$$



constant is analyt.

Cocontraction and coweakening make $!A$ a commutative comonoid.
Together they make $!A$ a commutative bialgebra.

Costructural rules

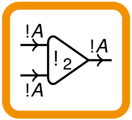
!'s functoriality, codiagonal, coterminal object, exponential isomorphism:

$$\nabla : A \ast A \to A$$

$$\rightsquigarrow$$

$$m : {!A} \otimes {!A} \cong {!(A \ast A)} \xrightarrow{!\nabla} {!A}$$

$$\rightsquigarrow$$



$$\leftarrow\!\!\rightsquigarrow$$

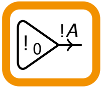
$f(x)$ analyt. $\Longrightarrow$ $f(x+y)$ analyt.

$$\top \to A$$

$$\rightsquigarrow$$

$$u : 1 \cong {!\top} \to {!A}$$

$$\rightsquigarrow$$



$$\leftarrow\!\!\rightsquigarrow$$

$f(x)$ analyt. $\Longrightarrow$ $f(0)$ analyt.

Cocontraction and coweakening make $!A$ a commutative monoid.

Together they make $!A$ a commutative bialgebra.

Costructural rules

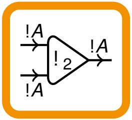
!'s functoriality, codiagonal, coterminal object, exponential isomorphism:

$$\nabla : A \ast A \to A$$

$$\Downarrow$$

$$m : !A \otimes !A \cong !(A \ast A) \xrightarrow{!\nabla} !A$$

$$\Downarrow$$



$$\Uparrow$$

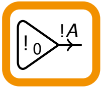
$f(x)$ analyt. $\implies$ $f(x+y)$ analyt.

$$\top \to A$$

$$\Downarrow$$

$$u : 1 \cong !\top \to !A$$

$$\Downarrow$$

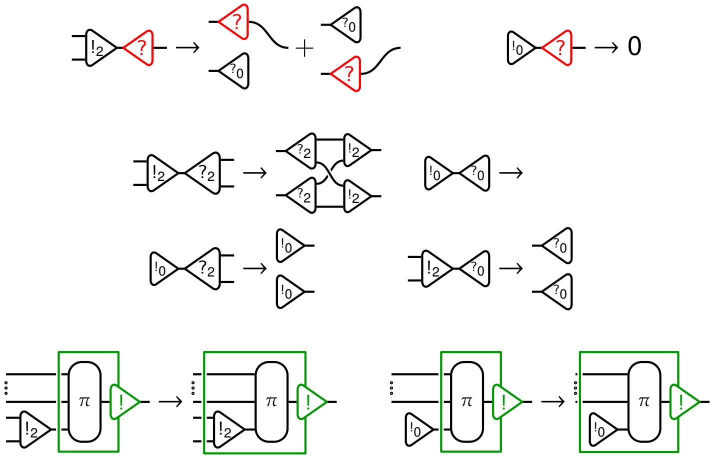


$$\Uparrow$$

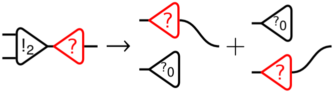
$f(x)$ analyt. $\implies$ $f(0)$ analyt.

Cocontraction and coweakening make $!A$ a commutative monoid.
Together they make $!A$ a commutative bialgebra.

New equalities/reductions



New equalities/reductions



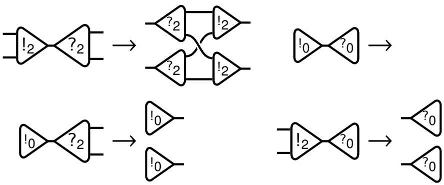
by definition (+ diagrams. . .)

$x + y$ and 0 as analytic functions

a query meets choice (or no choice)

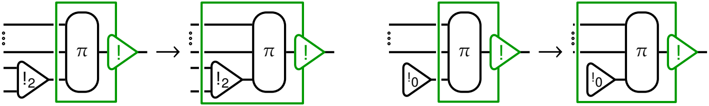
New equalities/reductions

- bialgebraic structure of $!A$
- commutation of sharing and sum
- nondeterministic routing

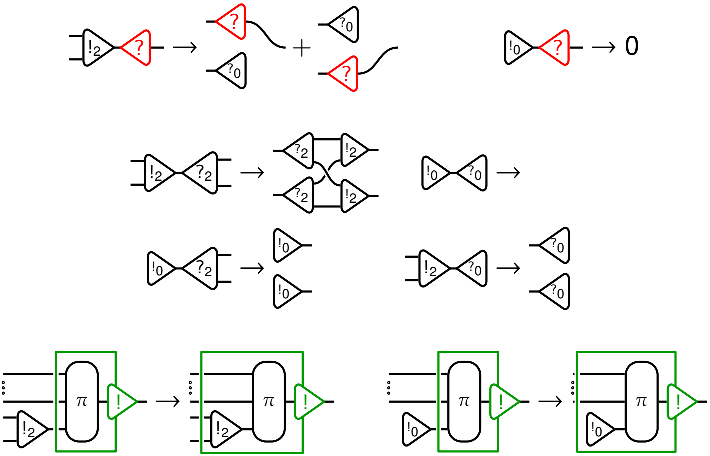


New equalities/reductions

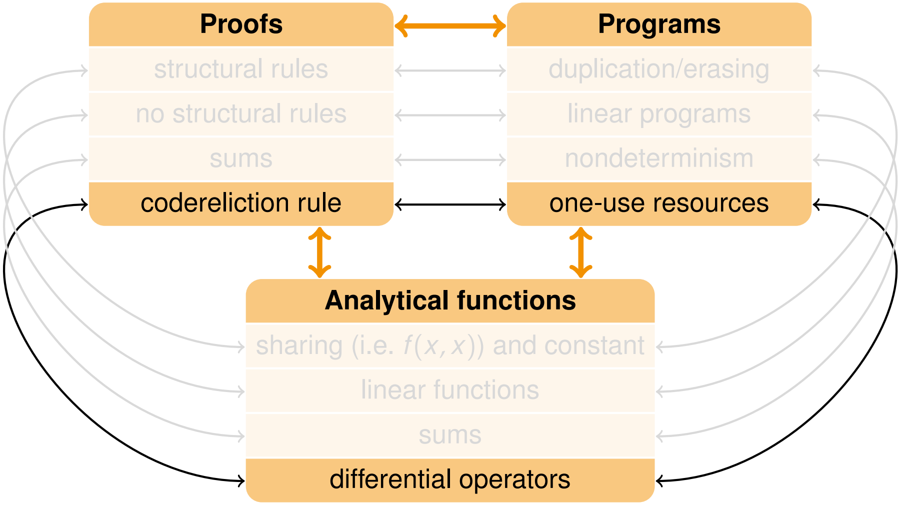
cocontraction and coweakening are as boxes



New equalities/reductions



Adding derivation

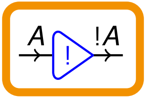


Codereliction

Symmetric to dereliction:

$$\eta : A \to\, !A$$

$\rightsquigarrow$

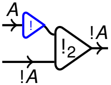


$\leftsquigarrow$

turning $f(x)$ into the linear map $\left.\dfrac{\partial f}{\partial x}\right|_{x=0}$.

Derivation in 0 is all that's needed:

$\frac{\partial f}{\partial x} = \left.\frac{\partial f(y+x)}{\partial y}\right|_{y=0}$, i.e.



.

Derivation in computation

Repetita iuvant…

Q: What is the derivative of a function?

A: Its best linear aproximation.

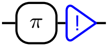
Q: What is the derivative of a program?

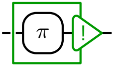
A: Its best linear aproximation!

.i.e. the (nondeterministic) aproximation using its input exactly once

⇝

One-use resources!

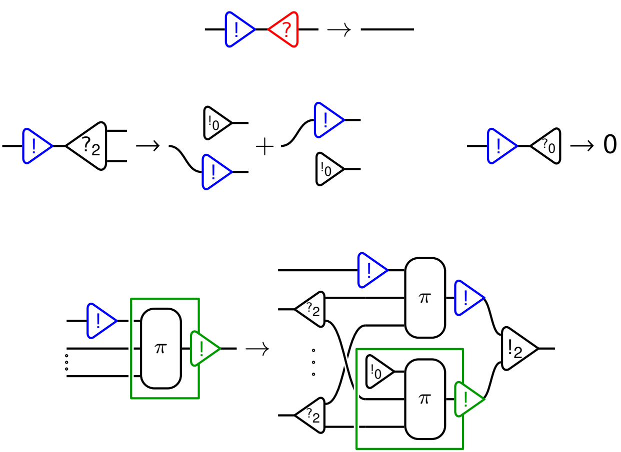


can go where


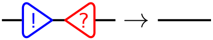
can, but won't be duplicated.

(or erased)

The new reductions



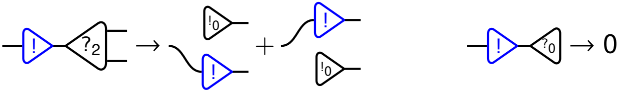
The new reductions



codereliction is right inverse of dereliction

$$\frac{\partial x}{\partial x}\Big|_{x=0} = \mathrm{id}$$

a query meets a one-use resource and is answered



...

$$\frac{\partial f(x,x)}{\partial x}\Big|_{x=0} = \frac{\partial f(y,z)}{\partial(y,z)}\Big|_{(y,z)=(0,0)} \cdot \frac{\partial(x,x)}{\partial x}\Big|_{x=0} = \frac{\partial f(x,0)}{\partial x}\Big|_{x=0} + \frac{\partial f(0,x)}{\partial x}\Big|_{x=0}, \qquad \frac{\partial C}{\partial x} = 0$$

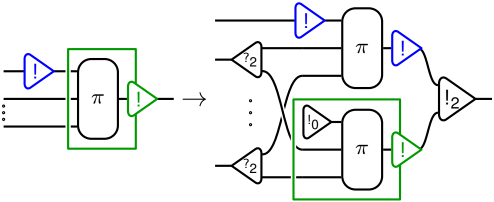
a one-use resource is contended by multiple (or no) queries

The new reductions

. . .

$$\left.\frac{\partial f(g(x))}{\partial x}\right|_{x=0} = \left.\frac{\partial f(y)}{\partial y}\right|_{y=g(0)} \cdot \left.\frac{\partial g(x)}{\partial x}\right|_{x=0},$$

a one-use resource is asked by a reusable one,
of which exactly one copy gets it

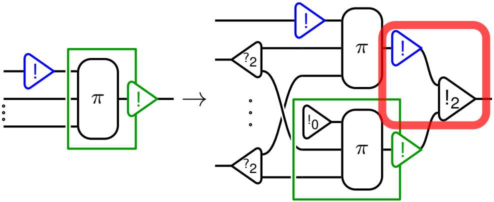


The new reductions

...

$$\frac{\partial f(g(x))}{\partial x}\Big|_{x=0} = \frac{\partial f(y)}{\partial y}\Big|_{y=g(0)} \cdot \frac{\partial g(x)}{\partial x}\Big|_{x=0},$$

a one-use resource is asked by a reusable one, of which exactly one copy gets it

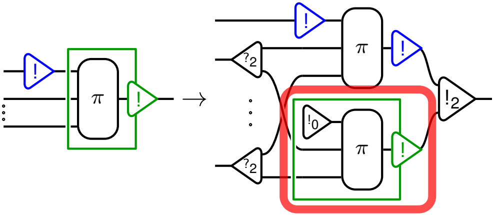


The new reductions

...

$$\frac{\partial f(g(x))}{\partial x}\Big|_{x=0} = \frac{\partial f(y)}{\partial y}\Big|_{y=g(0)} \cdot \frac{\partial g(x)}{\partial x}\Big|_{x=0},$$

a one-use resource is asked by a reusable one, of which exactly one copy gets it

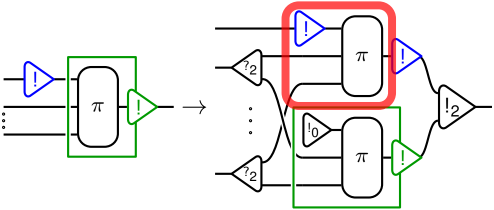


The new reductions

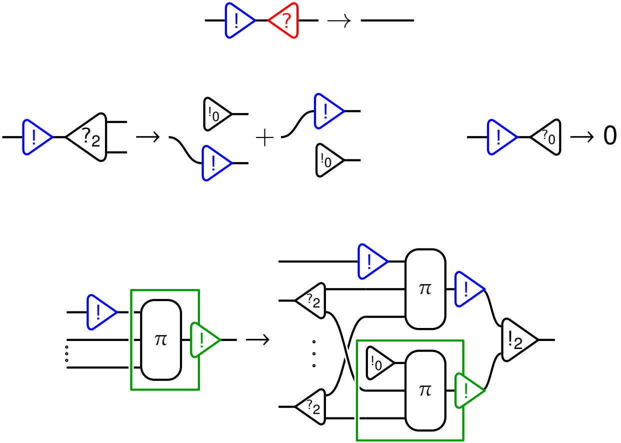
. . .

$$\frac{\partial f(g(x))}{\partial x}\Big|_{x=0} = \frac{\partial f(y)}{\partial y}\Big|_{y=g(0)} \cdot \frac{\partial g(x)}{\partial x}\Big|_{x=0},$$

a one-use resource is asked by a reusable one,
of which exactly one copy gets it



The new reductions



Are differential nets a good rewriting system?

Confluence? (aka Church-Rosser)

Is nondeterminism truly internal?
Do the possible outcomes depend on the reduction strategy?

Finite developments?

If taking a snapshot of possible reductions of a net and we reduce those ones only (and their copies thereof), does this terminate?

Termination?

Do all nondeterministic branches terminate?

Conservation?

Do non-erasing reductions preserve potential infinite reductions?

Standardization?

Can all reduction chains be rearranged in increasing depth?

Are differential nets a good rewriting system?

Confluence? (aka Church-Rosser)

Is nondeterminism truly internal?
Do the possible outcomes depend on the reduction strategy?

Finite developments?

If taking a snapshot of possible reductions of a net and we reduce those ones only (and their copies thereof), does this terminate?

Termination?

Do all nondeterministic branches terminate?

Conservation?

Do non-erasing reductions preserve potential infinite reductions?

Standardization?

Can all reduction chains be rearranged in increasing depth?

Summary of results

	FD	CR	WN	Cons.	Stand.	SN
Untyped DiLL						
Second order DiLL						
Propositional DiLL						

FD : finite developments
CR : Church Rosser
Cons. : conservation
Stand. : standardization

[Tr09] P. T.
Confluence of pure differential nets with promotion.
CSL'09, volume 5771 of *LNCS*, pages 500–514. 2009.

[PaTr09] Michele Pagani, P. T.
The conservation theorem for differential nets with promotion.
Accepted for publication.

[Pag09] Michele Pagani.
The cut-elimination theorem for differential nets with promotion.
TLCA'09, volume 5608 of *LNCS*, pages 219–233. 2009.

Summary of results

	FD	CR	WN	Cons.	Stand.	SN
Untyped DiLL	[Tr09]	[Tr09]	No	[PaTr09]	[PaTr09]	No
Second order DiLL	$\Downarrow$ Yes	$\Downarrow$ Yes	**?**	$\Downarrow$ Yes	$\Downarrow$ Yes	**?**
Propositional DiLL	$\Downarrow$ Yes	$\Downarrow$ Yes	[Pa09]	$\Downarrow$ Yes	$\Downarrow$ Yes	[PaTr09]

FD : finite developments
CR : Church Rosser
Cons. : conservation
Stand. : standardization

[Tr09] P. T.
Confluence of pure differential nets with promotion.
CSL'09, volume 5771 of *LNCS*, pages 500–514. 2009.

[PaTr09] Michele Pagani, P. T.
The conservation theorem for differential nets with promotion.
Accepted for publication.

[Pag09] Michele Pagani.
The cut-elimination theorem for differential nets with promotion.
TLCA'09, volume 5608 of *LNCS*, pages 219–233. 2009.

Summary of results

	FD	CR	WN	Cons.	Stand.	SN
Untyped DiLL	[Tr09]	[Tr09]	No	[PaTr09]	[PaTr09]	No
Second order DiLL	⇓ Yes	⇓ Yes	?	⇓ Yes	⇓ Yes	?
Propositional DiLL	⇓ Yes	⇓ Yes	[Pa09]	⇓ Yes	⇓ Yes	[PaTr09]

FD : finite developments
CR : Church Rosser
Cons. : conservation
Stand. : standardization

[Tr09] P. T.
Confluence of pure differential nets with promotion.
CSL'09, volume 5771 of *LNCS*, pages 500–514. 2009.

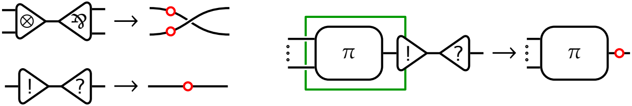
[Pag09] Michele Pagani.
The cut-elimination theorem for differential nets with promotion.
TLCA'09, volume 5608 of *LNCS*, pages 219–233. 2009.

[PaTr09] Michele Pagani, P. T.
The conservation theorem for differential nets with promotion.
Accepted for publication.

Stating the finite developments theorem

DiLL°: the system blocking "new" redexes

- "New" redexes are blocked via redefining reductions:



Theorem (Finite developments)

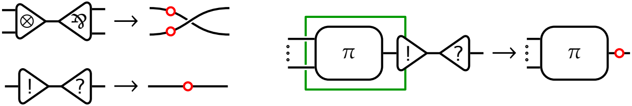
Untyped DiLL° *is strongly normalizing modulo.*

(In a truly untyped setting, more details are needed for clashes)

Stating the finite developments theorem

DiLL$^\circ$: the system blocking "new" redexes

- "New" redexes are blocked via redefining reductions:



Theorem (Finite developments)

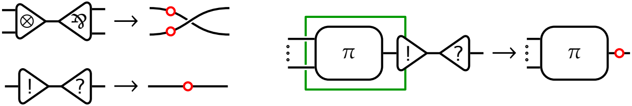
Untyped DiLL$^\circ$ *is strongly normalizing modulo.*

(In a truly untyped setting, more details are needed for clashes)

Stating the finite developments theorem

DiLL$^\circ$: the system blocking "new" redexes

- "New" redexes are blocked via redefining reductions:



Theorem (Finite developments)

Untyped DiLL$^\circ$ *is strongly normalizing modulo.*

(In a truly untyped setting, more details are needed for clashes)

Summary of results

	FD	CR	WN	Cons.	Stand.	SN
Untyped DiLL	[Tr09]	[Tr09]	No	[PaTr09]	[PaTr09]	No
Second order DiLL	⇓ Yes	⇓ Yes	?	⇓ Yes	⇓ Yes	?
Propositional DiLL	⇓ Yes	⇓ Yes	[Pa09]	⇓ Yes	⇓ Yes	[PaTr09]

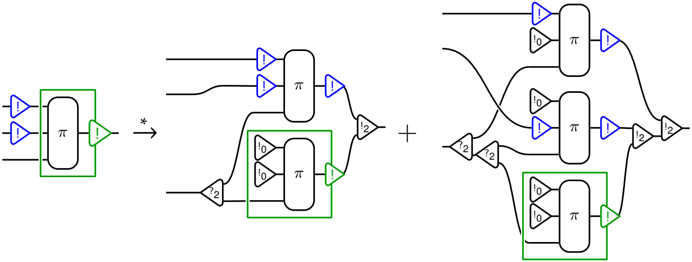
FD : finite developments
CR : Church Rosser
Cons. : conservation
Stand. : standardization

[Tr09] P. T.
Confluence of pure differential nets with promotion.
CSL'09, volume 5771 of *LNCS*, pages 500–514. 2009.

[PaTr09] Michele Pagani, P. T.
The conservation theorem for differential nets with promotion.
Accepted for publication.

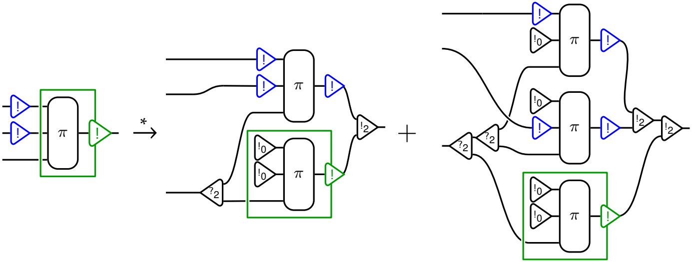
[Pag09] Michele Pagani.
The cut-elimination theorem for differential nets with promotion.
TLCA'09, volume 5608 of *LNCS*, pages 219–233. 2009.

We need associativity and neutrality



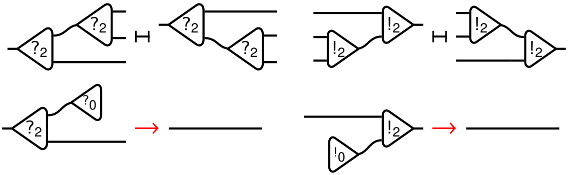
- Contractions and cocontractions need to be swapped to have confluence (associativity).
- Other confluence diagrams require merging (co)weakenings with (co)contractions (neutrality).

We need associativity and neutrality



- Contractions and cocontractions need to be swapped to have confluence (associativity).
- Other confluence diagrams require merging (co)weakenings with (co)contractions (neutrality).

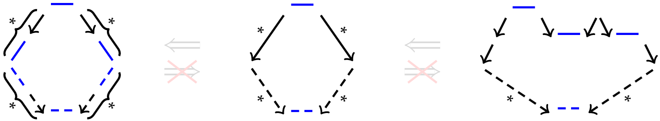
Associative equivalence and neutral reduction



- associative equivalence $\sim = \mapsto^*$
- neutral reduction (if reversed arbitrary (co)contraction trees can be generated)
- other (optional) equivalences can be added...

Reduction modulo equivalence

- Confluence properties in presence of an equivalence relation $\sim$:



confluence of $\to / \sim \; = \; \sim\to\sim$

confluence modulo

Church-Rosser modulo (CR$\sim$)
$(\to \cup \leftarrow \cup \sim)^* \subseteq \xrightarrow{*}\sim\xleftarrow{*}$

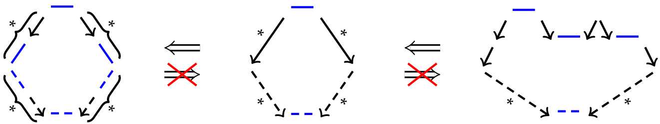
- Strong normalization modulo (SN$\sim$) is defined by SN of $\to/\sim \; = \; \sim\to\sim$.

Theorem (Confluence)

Untyped DiLL *is Church-Rosser modulo.*

Reduction modulo equivalence

- Confluence properties in presence of an equivalence relation $\sim$:



confluence of
$\to / \sim \; = \; \sim\to\sim$

confluence
modulo

Church-Rosser modulo
(CR$\sim$)
$(\to \cup \leftarrow \cup \sim)^* \subseteq \xrightarrow{*}\sim\xleftarrow{*}$

- Strong normalization modulo (SN$\sim$) is defined by SN of $\to/\sim \; = \; \sim\to\sim$.

Theorem (Confluence)

Untyped DiLL *is Church-Rosser modulo.*

Summary of results

	FD	CR	WN	Cons.	Stand.	SN
Untyped DiLL	[Tr09]	[Tr09]	No	[PaTr09]	[PaTr09]	No
Second order DiLL	$\Downarrow$ Yes	$\Downarrow$ Yes	**?**	$\Downarrow$ Yes	$\Downarrow$ Yes	**?**
Propositional DiLL	$\Downarrow$ Yes	$\Downarrow$ Yes	[Pa09]	$\Downarrow$ Yes	$\Downarrow$ Yes	[PaTr09]

FD : finite developments
CR : Church Rosser
Cons. : conservation
Stand. : standardization

[Tr09] P. T.
Confluence of pure differential nets with promotion.
CSL'09, volume 5771 of *LNCS*, pages 500–514. 2009.

[PaTr09] Michele Pagani, P. T.
The conservation theorem for differential nets with promotion.
Accepted for publication.

[Pag09] Michele Pagani.
The cut-elimination theorem for differential nets with promotion.
TLCA'09, volume 5608 of *LNCS*, pages 219–233. 2009.

Summary of results

	FD	CR	WN	Cons.	Stand.	SN
Untyped DiLL	[Tr09]	[Tr09]	No	[PaTr09]	[PaTr09]	No
Second order DiLL	⇓ Yes	⇓ Yes	?	⇓ Yes	⇓ Yes	?
Propositional DiLL	⇓ Yes	⇓ Yes	[Pa09]	⇓ Yes	⇓ Yes	[PaTr09]

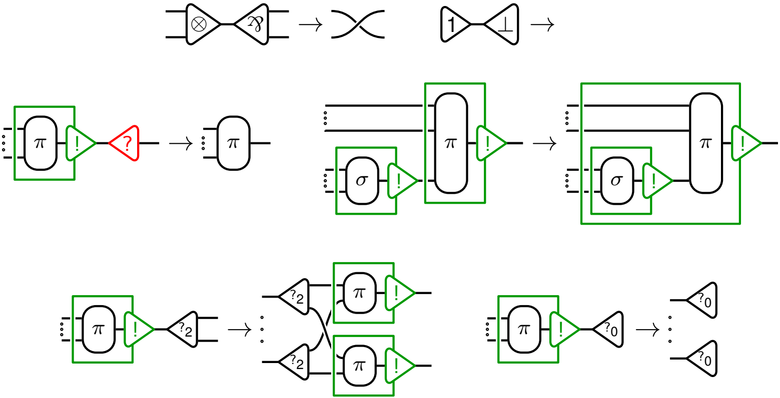
FD : finite developments
CR : Church Rosser
Cons. : conservation
Stand. : standardization

[Tr09] P. T.
Confluence of pure differential nets with promotion.
CSL'09, volume 5771 of *LNCS*, pages 500–514. 2009.

[Pag09] Michele Pagani.
The cut-elimination theorem for differential nets with promotion.
TLCA'09, volume 5608 of *LNCS*, pages 219–233. 2009.

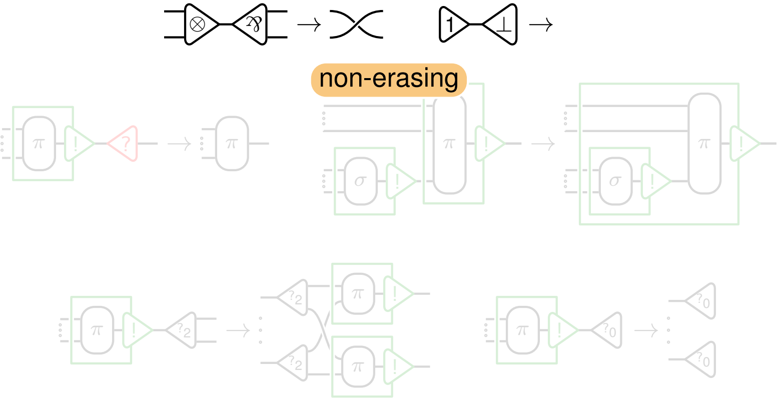
[PaTr09] Michele Pagani, P. T.
The conservation theorem for differential nets with promotion.
Accepted for publication.

From equalities. . . to rewriting



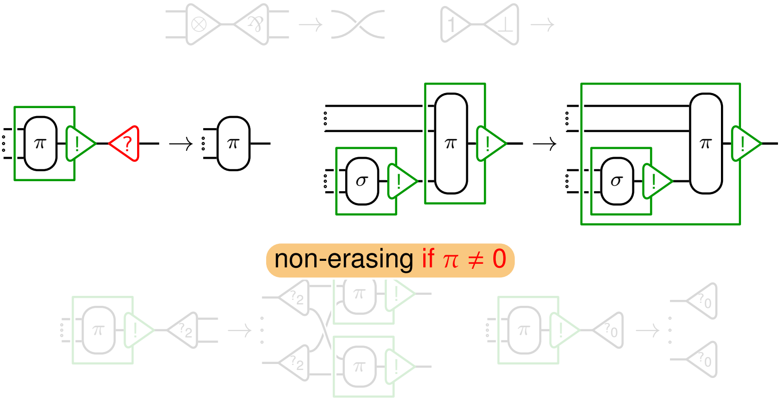
from string diagrams. . . to interaction nets!

From equalities. . . to rewriting



from string diagrams. . . to interaction nets!

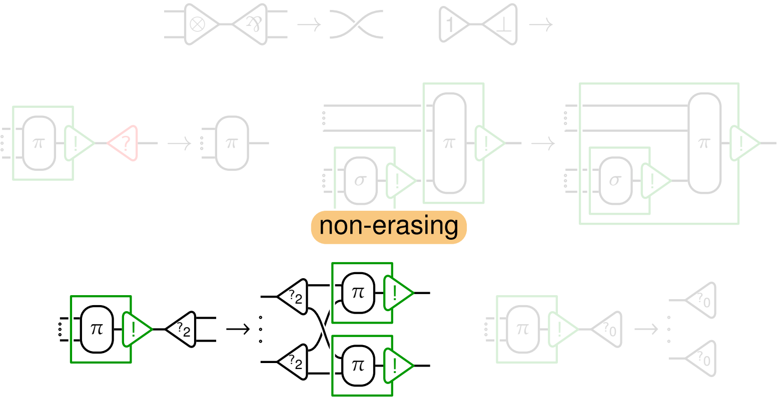
From equalities. . . to rewriting



non-erasing if $\pi \neq 0$

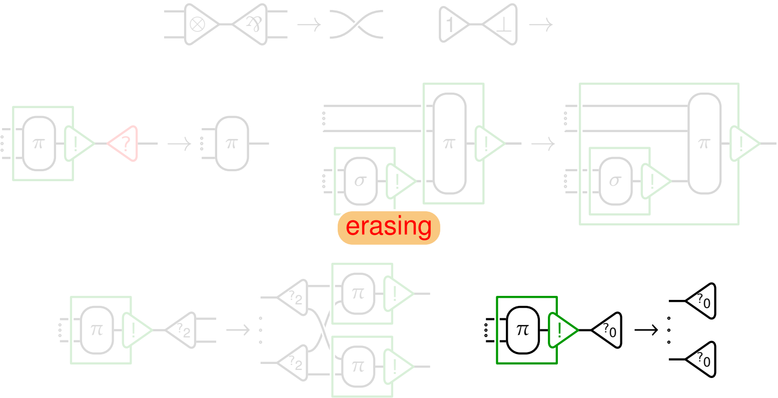
from string diagrams. . . to interaction nets!

From equalities. . . to rewriting



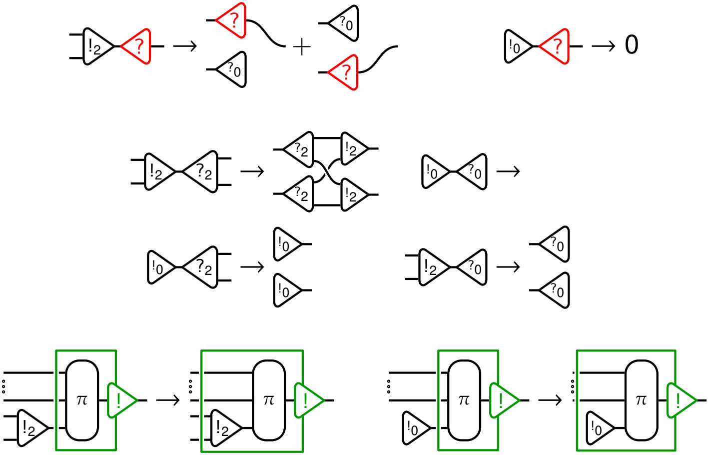
from string diagrams. . . to interaction nets!

From equalities. . . to rewriting

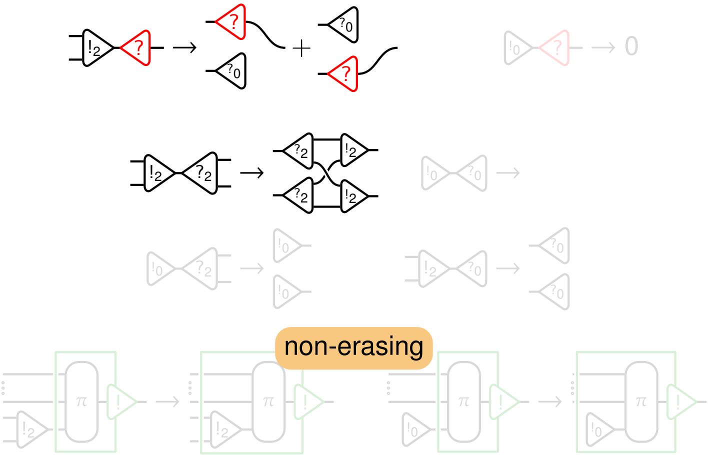


from string diagrams. . . to interaction nets!

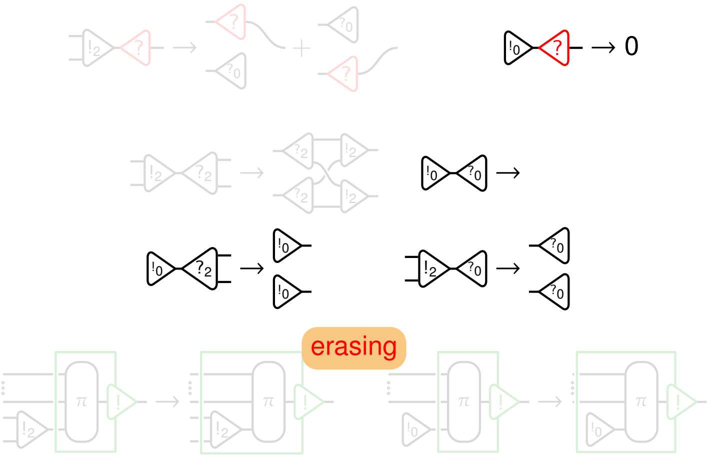
New equalities/reductions



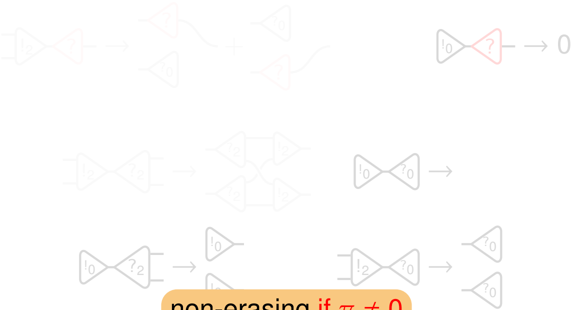
New equalities/reductions



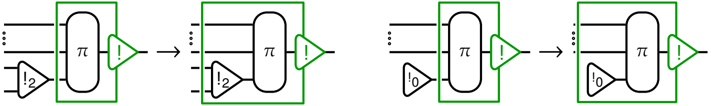
New equalities/reductions



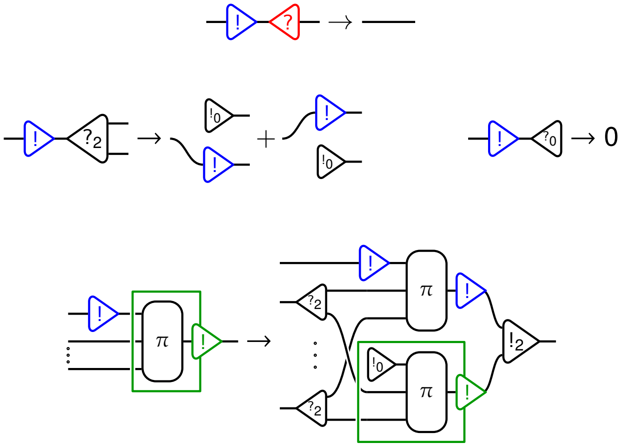
New equalities/reductions



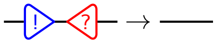
non-erasing if $\pi \neq 0$

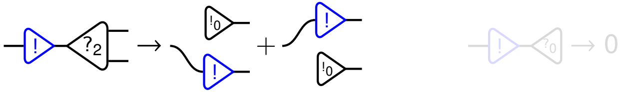


The new reductions



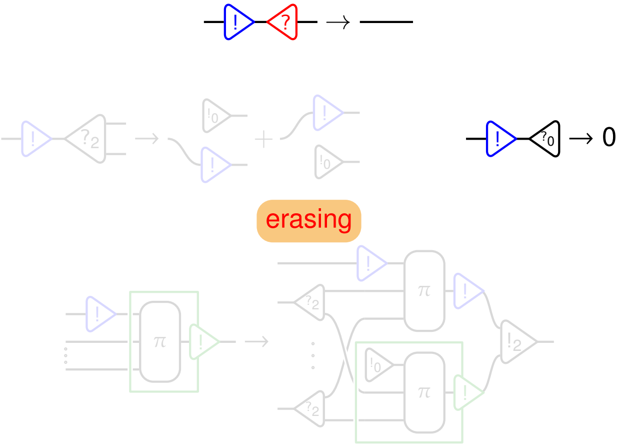
The new reductions





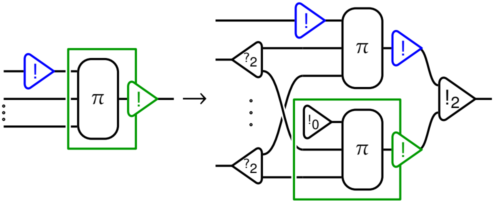
non-erasing

The new reductions



The new reductions

non-erasing if $\pi \neq 0$



Two problems

We want:

Theorem

If $\pi \xrightarrow{\neg er} \sigma$ *and* $\pi \notin \mathrm{SN}\sim$ *then* $\sigma \notin \mathrm{SN}\sim$.

1. In the purely untyped case, this is false (contrary to LL)
2. Even amending that, the standard proof needs local confluence of $\neg$er, which is false

Two problems

We want:

Theorem

If $\pi \in \mathrm{WN}_{\neg er}$ *then* $\pi \in \mathrm{SN} \sim$.

1. In the purely untyped case, this is false (contrary to LL)
2. Even amending that, the standard proof needs local confluence of $\neg$er, which is false

Two problems

We want:

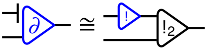
Theorem

If $\pi \in \mathrm{WN}_{\neg er}$ *then* $\pi \in \mathrm{SN} \sim$.

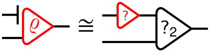
1. In the purely untyped case, this is false (contrary to LL)
2. Even amending that, the standard proof needs local confluence of $\neg$er, which is false

Enter $\mathrm{DiLL}_{\partial\varrho}$

- Asymmetry between dereliction and codereliction. . .
- . . . what if we switch to partial derivation as primitive, with a symmetric counterpart for dereliction?

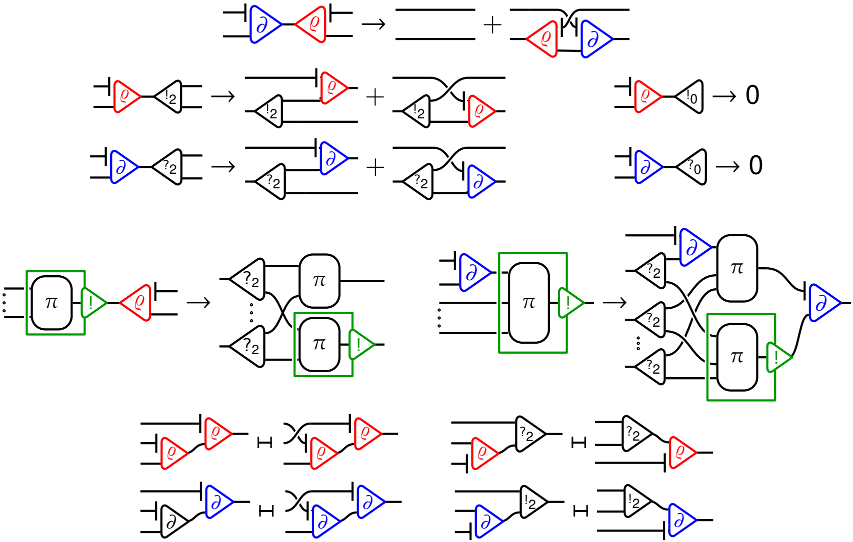


linear substitution



linear query

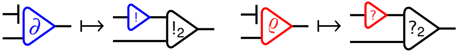
DiLL$_{\partial\varrho}$



Properties of $\mathrm{DiLL}_{\partial\varrho}$

- It enjoys confluence (both for regular and non-erasing)
- It enjoys conservation
- As already seen, there is the translation

$\mathrm{DiLL} \xrightarrow{(-)^{\vartriangle}} \mathrm{DiLL}_{\partial\varrho}$:



- $\pi^{\vartriangle}$ simulates π (if $\pi \leadsto \sigma$ then $\pi^{\vartriangle} \leadsto \sigma^{\vartriangle}$)
- Moreover, if π is $\mathrm{WN}_{\neg\mathrm{er}}$ then so is $\pi^{\vartriangle}$ (not trivial!)

$$\pi \in \mathrm{WN}_{\neg\mathrm{er}} \implies \pi^{\vartriangle} \in \mathrm{WN}_{\neg\mathrm{er}} \implies \pi^{\vartriangle} \in \mathrm{SN}_{\sim} \implies \pi \in \mathrm{SN}_{\sim}$$

So conservation holds for both DiLL and $\mathrm{DiLL}_{\partial\varrho}$

Summary of results

	FD	CR	WN	Cons.	Stand.	SN
Untyped DiLL	[Tr09]	[Tr09]	No	[PaTr09]	[PaTr09]	No
Second order DiLL	⇓ Yes	⇓ Yes	?	⇓ Yes	⇓ Yes	?
Propositional DiLL	⇓ Yes	⇓ Yes	[Pa09]	⇓ Yes	⇓ Yes	[PaTr09]

FD : finite developments
CR : Church Rosser
Cons. : conservation
Stand. : standardization

[Tr09] P. T.
Confluence of pure differential nets with promotion.
CSL'09, volume 5771 of *LNCS*, pages 500–514. 2009.

[Pag09] Michele Pagani.
The cut-elimination theorem for differential nets with promotion.
TLCA'09, volume 5608 of *LNCS*, pages 219–233. 2009.

[PaTr09] Michele Pagani, P. T.
The conservation theorem for differential nets with promotion.
Accepted for publication.

Summary of results

	FD	CR	WN	Cons.	Stand.	SN
Untyped DiLL	[Tr09]	[Tr09]	No	[PaTr09]	[PaTr09]	No
Second order DiLL	$\Downarrow$ Yes	$\Downarrow$ Yes	?	$\Downarrow$ Yes	$\Downarrow$ Yes	?
Propositional DiLL	$\Downarrow$ Yes	$\Downarrow$ Yes	[Pa09]	$\Downarrow$ Yes	$\Downarrow$ Yes	[PaTr09]

FD : finite developments
CR : Church Rosser
Cons. : conservation
Stand. : standardization

[Tr09] P. T.
Confluence of pure differential nets with promotion.
CSL'09, volume 5771 of *LNCS*, pages 500–514. 2009.

[Pag09] Michele Pagani.
The cut-elimination theorem for differential nets with promotion.
TLCA'09, volume 5608 of *LNCS*, pages 219–233. 2009.

[PaTr09] Michele Pagani, P. T.
The conservation theorem for differential nets with promotion.
Accepted for publication.

From conservation to SN

- Conservation theorem allows to transform a WN proof into an SN one
- Propositional DiLL has WN $\implies$ propositional DiLL and $\mathrm{DiLL}_{\partial\varrho}$ are SN
- Second order DiLL has WN (candidates of reducibility), but result not published yet (but trust me)

Conclusion: DiLL and $\mathrm{DiLL}_{\partial\varrho}$ share LL proof nets' good rewriting properties!

From conservation to SN

- Conservation theorem allows to transform a WN proof into an SN one
- Propositional DiLL has WN $\implies$ propositional DiLL and $\text{DiLL}_{\partial\varrho}$ are SN
- Second order DiLL has WN (candidates of reducibility), but result not published yet (but trust me)

Conclusion: DiLL and $\text{DiLL}_{\partial\varrho}$ share LL proof nets' good rewriting properties!

Questions?

. . . if I did not make you sleep. . .