CRITICAL PAIRS IN 2-DIMENSIONAL REWRITING SYSTEMS

SAMUEL MIMRAM

OPERADS AND REWRITING

3 NOVEMBER 2011

TOWARDS HIGHER-DIMENSIONAL REWRITING THEORY

Rewriting theory has proven to be very useful to study

- monoids (and groups)
- term algebras

TOWARDS HIGHER-DIMENSIONAL REWRITING THEORY

Rewriting theory has proven to be very useful to study

- monoids (and groups)
- term algebras
- ***n*-categories?**

`Can we generalize it`
`to higher dimensions?`

TOWARDS HIGHER-DIMENSIONAL REWRITING THEORY

Rewriting theory has proven to be very useful to study

- monoids (and groups)
- term algebras
- ***n*-categories?**

`Can we generalize it to higher dimensions?`

In this talk, I will be interested in extending the procedures of **unification** in higher dimensions.

TOWARDS HIGHER-DIMENSIONAL REWRITING THEORY

Rewriting theory has proven to be very useful to study

- monoids (and groups)
- term algebras
- ***n*-categories?**

`Can we generalize it to higher dimensions?`

In this talk, I will be interested in extending the procedures of **unification** in dimension 2.

REWRITING SYSTEMS

REWRITING SYSTEMS

A **rewriting system** consists of

- a set of *terms* generated by a free construction:
  - free monoid: *string rewriting systems*
  - free term algebra: *term rewriting systems*
- a set of *rewriting rules*: $r : t \to u$

Example

$\Sigma = \{a, b\}$ $\qquad$ $\text{terms} = \Sigma^*$ $\qquad$ $\text{rules} = \{ba \to ab\}$

REWRITING SYSTEMS

A **rewriting system** consists of

- a set of *terms* generated by a free construction:
  - free monoid: *string rewriting systems*
  - free term algebra: *term rewriting systems*
- a set of *rewriting rules*: $r : t \to u$

A term t **rewrites** to a term t' when there exists

- a rule $r : u \to u'$
- a context C such that $t = C[u]$ and $t' = C[u']$

Example

$\Sigma = \{a, b\}$ $\quad$ $\text{terms} = \Sigma^*$ $\quad$ $\text{rules} = \{ba \to ab\}$

$$aa\color{red}{ba}ab \xrightarrow{aa\color{red}{r}ab} aa\color{red}{ab}ab$$

CONVERGENT REWRITING SYSTEMS

- A rewriting system can be **terminating** when there is no infinite reduction path

$$\begin{array}{c} t \\ \downarrow \\ t_1 \\ \downarrow \\ t_2 \\ \downarrow \\ \vdots \end{array}$$

CONVERGENT REWRITING SYSTEMS

- A rewriting system can be **terminating**
- A rewriting can be **confluent** when

$$\begin{tikzcd}
 & t \arrow[dl, "*"'] \arrow[dr, "*"] & \\
u & & v
\end{tikzcd}$$

CONVERGENT REWRITING SYSTEMS

- A rewriting system can be **terminating**
- A rewriting can be **confluent** when

$$
\begin{tikzcd}
& t \arrow[dl, "*"'] \arrow[dr, "*"] & \\
u \arrow[dr, dotted, "*"'] & & v \arrow[dl, dotted, "*"] \\
& w &
\end{tikzcd}
$$

CONVERGENT REWRITING SYSTEMS

- A rewriting system can be **terminating**
- A rewriting can be **confluent** when

$$
\begin{tikzcd}
& t \arrow[dl, "*"'] \arrow[dr, "*"] & \\
u \arrow[dr, dotted, "*"'] & & v \arrow[dl, dotted, "*"] \\
& w &
\end{tikzcd}
$$

- A rewriting system is **convergent**
  when both terminating and confluent

CONVERGENT REWRITING SYSTEMS

- A rewriting system can be **terminating**
- A rewriting can be **confluent** when

$$\begin{tikzcd}
 & t \arrow[dl, "*"'] \arrow[dr, "*"] & \\
u \arrow[dr, dotted, "*"'] & & v \arrow[dl, dotted, "*"] \\
 & w &
\end{tikzcd}$$

- A rewriting system is **convergent**
  when both terminating and confluent

In a convergent rewriting system, every term has a **normal form**:
canonical representative of terms modulo rewriting.

Why
are those properties
interesting?

PRESENTATIONS OF MONOIDS

A **presentation**

$$\langle G \mid R \rangle$$

of a monoid M consists of

- a set G of *generators*
- a set $R \subseteq G^* \times G^*$ of *relations*

such that

$$M \qquad \cong \qquad G^* / \equiv_R$$

Example

- $\mathbb{N} \cong \langle a \mid \rangle$
- $\mathbb{N}/2\mathbb{N} \cong \langle a \mid aa = 1 \rangle$
- $\mathbb{N} \times \mathbb{N} \cong \langle a, b \mid ba = ab \rangle$
- $\mathfrak{S}_n \cong \langle \sigma_1, \ldots, \sigma_n \mid \sigma_i \sigma_{i+1} \sigma_i = \sigma_{i+1} \sigma_i \sigma_{i+1},\ \sigma_i^2 = 1,\ \sigma_i \sigma_j = \sigma_j \sigma_i \rangle$
- ...

PRESENTATIONS OF MONOIDS

How do we show that $M \cong \langle G \mid R \rangle$ i.e. $M \cong G^* / \equiv_R$?

PRESENTATIONS OF MONOIDS

How do we show that $M \cong \langle G \mid R \rangle$ i.e. $M \cong G^* / \equiv_R$?

1. Orient R to get a string rewriting system.
2. Show that the rewriting system is terminating.
3. Show that the rewriting system is confluent.
4. Show that the normal forms are in bijection with M.

PRESENTATIONS OF MONOIDS

How do we show that $M \cong \langle G \mid R \rangle$ i.e. $M \cong G^* / \equiv_R$?

1. Orient R to get a string rewriting system.
2. Show that the rewriting system is terminating.
3. Show that the rewriting system is confluent.
4. Show that the normal forms are in bijection with M.

Example $\mathbb{N} \times (\mathbb{N}/2\mathbb{N}) \overset{?}{\cong} \langle a, b \mid ba = ab,\ bb = 1 \rangle$

PRESENTATIONS OF MONOIDS

How do we show that $M \cong \langle G \mid R \rangle$ i.e. $M \cong G^* / \equiv_R$?

1. Orient R to get a string rewriting system.
2. Show that the rewriting system is terminating.
3. Show that the rewriting system is confluent.
4. Show that the normal forms are in bijection with M.

Example $\mathbb{N} \times (\mathbb{N}/2\mathbb{N}) \quad \overset{?}{\cong} \quad \langle a, b \mid ba \to ab,\ bb \to 1 \rangle$

PRESENTATIONS OF MONOIDS

How do we show that $M \cong \langle G \mid R \rangle$ i.e. $M \cong G^* / \equiv_R$?

1. Orient R to get a string rewriting system.
2. Show that the rewriting system is terminating.
3. Show that the rewriting system is confluent.
4. Show that the normal forms are in bijection with M.

Example $\mathbb{N} \times (\mathbb{N}/2\mathbb{N}) \overset{?}{\cong} \langle a, b \mid ba \to ab,\ bb \to 1 \rangle$

Normal forms are:

a^n and $a^n b$

They are in bijection with $\mathbb{N} \times (\mathbb{N}/2\mathbb{N})$!

How do we show
that a rewriting system
is confluent?

CRITICAL PAIRS

Given a *terminating* rewriting system the following are equivalent:

1. the rewriting system is **confluent**

$$
\begin{tikzcd}
& t \arrow[dl, "*"'] \arrow[dr, "*"] & \\
u \arrow[dr, dotted, "*"'] & & v \arrow[dl, dotted, "*"] \\
& w &
\end{tikzcd}
$$

CRITICAL PAIRS

Given a *terminating* rewriting system the following are equivalent:

1. the rewriting system is **confluent**
2. the rewriting system is **locally confluent**

$$
\begin{tikzcd}
 & t \arrow[dl] \arrow[dr] & \\
u \arrow[dr, dotted, "*"'] & & v \arrow[dl, dotted, "*"] \\
 & w &
\end{tikzcd}
$$

CRITICAL PAIRS

Given a *terminating* rewriting system the following are equivalent:

1. the rewriting system is **confluent**
2. the rewriting system is **locally confluent**

$$
\begin{tikzcd}
 & C[t] \arrow[dl] \arrow[dr] & \\
C[u] \arrow[dr, dotted, "*"'] & & C[v] \arrow[dl, dotted, "*"] \\
 & C[w] &
\end{tikzcd}
$$

CRITICAL PAIRS

Given a *terminating* rewriting system the following are equivalent:

1. the rewriting system is **confluent**
2. the rewriting system is **locally confluent**

$$\begin{tikzcd}
 & C[t] \arrow[dl] \arrow[dr] & \\
C[u] \arrow[dr, dotted, "*"'] & & C[v] \arrow[dl, dotted, "*"] \\
 & C[w] &
\end{tikzcd}$$

3. all the **critical pairs are joinable**
   i.e. the property above is satisfied for all minimal t

CRITICAL PAIRS

Given a <u>*terminating*</u> rewriting system the following are equivalent:

1. the rewriting system is **confluent**
2. the rewriting system is **locally confluent**
3. all the **critical pairs are joinable**
   i.e. the property above is satisfied for all minimal t

Example $\mathbb{N} \times (\mathbb{N}/2\mathbb{N}) \quad \overset{?}{\cong} \quad \langle a, b \mid ba \to ab,\ bb \to 1 \rangle$

Critical pairs are:

$$\begin{tikzcd} & bba \arrow[dl] \arrow[dr] & \\ a & & bab \end{tikzcd} \qquad \begin{tikzcd} & bbb \arrow[dl] \arrow[dr] & \\ b & & b \end{tikzcd}$$

CRITICAL PAIRS

Given a <u>*terminating*</u> rewriting system the following are equivalent:

1. the rewriting system is **confluent**
2. the rewriting system is **locally confluent**
3. all the **critical pairs are joinable**
   i.e. the property above is satisfied for all minimal t

Example $\mathbb{N} \times (\mathbb{N}/2\mathbb{N}) \quad \overset{?}{\cong} \quad \langle a, b \mid ba \to ab,\ bb \to 1 \rangle$

Critical pairs are joinable:

$$\begin{tikzcd}
 & bba \arrow[dl] \arrow[dr] & \\
a \arrow[dr, equal] & & bab \arrow[dl, "*"] \\
 & a &
\end{tikzcd}
\qquad\qquad
\begin{tikzcd}
 & bbb \arrow[dl] \arrow[dr] & \\
b \arrow[dr, equal] & & b \arrow[dl, equal] \\
 & b &
\end{tikzcd}$$

```
string rs  =  presentation of a monoid
  term rs  =  presentation of ?
```

TERM REWRITING SYSTEMS

- A **signature** (Σ, α) consists of
  - a set Σ of *generators*
  - an *arity* function $\alpha : \Sigma \to \mathbb{N}$
- Terms are elements of the free algebra Σ^* over this signature

Example

The TRS of commutative monoids: $\Sigma = \{m : 2,\ e : 0\}$

$$R = \left\{ \begin{array}{lrl} \alpha : & m(m(x,y),z) & \to m(x,m(y,z)) \\ \lambda : & m(e,x) & \to x \\ \rho : & m(x,e) & \to x \\ \gamma : & m(x,y) & \to m(y,x) \end{array} \right\}$$

PRESENTATIONS OF LAWVERE THEORIES

String rewriting systems correspond to presentations of monoids.

PRESENTATIONS OF LAWVERE THEORIES

Term rewriting systems correspond to presentations of **Lawvere theories**.

PRESENTATIONS OF LAWVERE THEORIES

A **Lawvere theory** is a category $\mathcal{C}$

- whose objects are integers
- which is cartesian
- whose cartesian product is given on objects by addition

PRESENTATIONS OF LAWVERE THEORIES

A **Lawvere theory** is a category $\mathcal{C}$

- whose objects are integers
- which is cartesian
- whose cartesian product is given on objects by addition

TRS $((\Sigma, \alpha), R)$ induce a LT whose morphisms $m \to n$ are n-uples of terms with variables $x_1, \ldots, x_m$, considered modulo relations.

Presentation: $\mathcal{C} \quad \cong \quad \Sigma^* / \equiv_R$

PRESENTATIONS OF LAWVERE THEORIES

A **Lawvere theory** is a category $\mathcal{C}$

- whose objects are integers
- which is cartesian
- whose cartesian product is given on objects by addition

TRS $((\Sigma, \alpha), R)$ induce a LT whose morphisms $m \to n$ are n-uples of terms with variables $x_1, \ldots, x_m$, considered modulo relations.

Presentation: $\mathcal{C} \quad \cong \quad \Sigma^* / \equiv_R$

Example

Consider the TRS of commutative monoids: $\Sigma = \{m : 2,\ e : 0\}$

$$R = \left\{ \begin{array}{rrcl} \alpha : & m(m(x,y),z) & \to & m(x, m(y,z)) \\ \lambda : & m(e,x) & \to & x \\ \rho : & m(x,e) & \to & x \\ \gamma : & m(x,y) & \to & m(y,x) \end{array} \right\}$$

PRESENTATIONS OF LAWVERE THEORIES

A **Lawvere theory** is a category $\mathcal{C}$

- whose objects are integers
- which is cartesian
- whose cartesian product is given on objects by addition

TRS $((\Sigma, \alpha), R)$ induce a LT whose morphisms $m \to n$ are n-uples of terms with variables $x_1, \ldots, x_m$, considered modulo relations.

Presentation: $\mathcal{C} \quad \cong \quad \Sigma^* / \equiv_R$

Example

Consider the TRS of commutative monoids: $\Sigma = \{m : 2,\ e : 0\}$

It presents the Lawvere theory whose morphisms $M : m \to n$ are $(m \times n)$-matrices with coefficients in $\mathbb{N}$.

PRESENTATIONS OF LAWVERE THEORIES

A **Lawvere theory** is a category $\mathcal{C}$

- whose objects are integers
- which is cartesian
- whose cartesian product is given on objects by addition

TRS $((\Sigma, \alpha), R)$ induce a LT whose morphisms $m \to n$ are n-uples of terms with variables $x_1, \ldots, x_m$, considered modulo relations.

Presentation: $\mathcal{C} \quad \cong \quad \Sigma^* / \equiv_R$

Example

Consider the TRS of commutative monoids: $\Sigma = \{m : 2,\ e : 0\}$

$$[m(m(x_1, x_1), x_2)\ ;\ e\ ;\ x_2] \quad : 2 \to 3$$

PRESENTATIONS OF LAWVERE THEORIES

A **Lawvere theory** is a category $\mathcal{C}$

- whose objects are integers
- which is cartesian
- whose cartesian product is given on objects by addition

TRS $((\Sigma, \alpha), R)$ induce a LT whose morphisms $m \to n$ are n-uples of terms with variables $x_1, \ldots, x_m$, considered modulo relations.

Presentation: $\mathcal{C} \quad \cong \quad \Sigma^* / \equiv_R$

Example

Consider the TRS of commutative monoids: $\Sigma = \{m : 2,\ e : 0\}$

$$[m(m(x_1, x_1), x_2)\ ;\ e\ ;\ x_2]\ \ : 2 \to 3 \qquad \rightsquigarrow \qquad \begin{pmatrix} 2 & 0 & 0 \\ 1 & 0 & 1 \end{pmatrix}$$

PRESENTATIONS OF LAWVERE THEORIES

A **Lawvere theory** is a category $\mathcal{C}$

- whose objects are integers
- which is cartesian
- whose cartesian product is given on objects by addition

TRS $((\Sigma, \alpha), R)$ induce a LT whose morphisms $m \to n$ are n-uples of terms with variables $x_1, \ldots, x_m$, considered modulo relations.

Presentation: $\mathcal{C} \quad \cong \quad \Sigma^* / \equiv_R$

Example

Consider the TRS of commutative monoids: $\Sigma = \{m : 2,\ e : 0\}$

$$\begin{array}{r} [m(m(x_1, x_1), x_2)\ ;\ e\ ;\ x_2] \\ [m(x_1, m(x_1, x_2))\ ;\ e\ ;\ m(e, x_2)] \end{array} : 2 \to 3 \qquad \rightsquigarrow \qquad \begin{pmatrix} 2 & 0 & 0 \\ 1 & 0 & 1 \end{pmatrix}$$

PRESENTATIONS OF LAWVERE THEORIES

A **Lawvere theory** is a category $\mathcal{C}$

- whose objects are integers
- which is cartesian
- whose cartesian product is given on objects by addition

TRS $((\Sigma, \alpha), R)$ induce a LT whose morphisms $m \to n$ are n-uples of terms with variables $x_1, \ldots, x_m$, considered modulo relations.

Presentation: $\mathcal{C} \quad \cong \quad \Sigma^* / \equiv_R$

Example

Consider the TRS of commutative monoids: $\Sigma = \{m : 2,\ e : 0\}$

$$\begin{array}{r} [m(m(x_1, x_1), x_2)\ ;\ e\ ;\ x_2] \\ [m(x_1, m(x_1, x_2))\ ;\ e\ ;\ m(e, x_2)] \end{array} : 2 \to 3 \qquad \rightsquigarrow \qquad \begin{pmatrix} 2 & 0 & 0 \\ 1 & 0 & 1 \end{pmatrix}$$

Use rewriting theory!

`Can we use the same techniques`
`in order to build`
`presentations of` n`-categories?`

POLYGRAPHS

[Street76,Burroni93,Power90]

PRESENTING n-CATEGORIES

We want to generalize rewriting systems

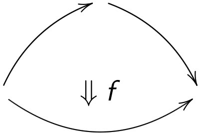
dimension	rewr. syst.	presents
1	string	monoid

$$\xrightarrow{a}\ \xrightarrow{b}\ \xrightarrow{c}$$

PRESENTING n-CATEGORIES

We want to generalize rewriting systems

dimension	rewr. syst.	presents
1	string	monoid
2	term	Lawvere th.



PRESENTING n-CATEGORIES

We want to generalize rewriting systems

dimension	rewr. syst.	presents
0	element	set
1	string	monoid
2	term	Lawvere th.

PRESENTING n-CATEGORIES

We want to generalize rewriting systems

dimension	rewr. syst.	presents
0	element	set
1	string	monoid
2	term	Lawvere th.

PRESENTING n-CATEGORIES

We want to generalize rewriting systems

dimension	rewr. syst.	presents
0	element	0-category
1	string	monoid
2	term	Lawvere th.

$$\text{set} \quad = \quad \text{0-category}$$

PRESENTING n-CATEGORIES

We want to generalize rewriting systems

dimension	rewr. syst.	presents
0	element	0-category
1	string	monoid
2	term	Lawvere th.

PRESENTING n-CATEGORIES

We want to generalize rewriting systems

dimension	rewr. syst.	presents
0	element	0-category
1	string	1-category
2	term	Lawvere th.

monoid $=$ 1-category with only one object

Generalization: $\xrightarrow{a} \xrightarrow{b} \quad \rightsquigarrow \quad x \xrightarrow{a} y \xrightarrow{b} y$

PRESENTING n-CATEGORIES

We want to generalize rewriting systems

dimension	rewr. syst.	presents
0	element	0-category
1	string	1-category
2	term	Lawvere th.

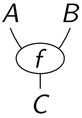
PRESENTING n-CATEGORIES

We want to generalize rewriting systems

dimension	rewr. syst.	presents
0	element	0-category
1	string	1-category
2	term	cartesian category

Lawvere th. $=$ cartesian category with $\mathbb{N}$ as objects

Generalization: 

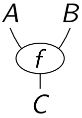
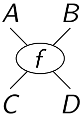
$\rightsquigarrow$ 

PRESENTING n-CATEGORIES

We want to generalize rewriting systems

dimension	rewr. syst.	presents
0	element	0-category
1	string	1-category
2	term	monoidal category

cartesian category $=$ monoidal category in which every object is a comonoid

Generalization:  $\rightsquigarrow$ 

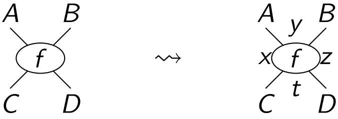
PRESENTING n-CATEGORIES

We want to generalize rewriting systems

dimension	rewr. syst.	presents
0	element	0-category
1	string	1-category
2	term	2-category

monoidal category $\quad = \quad$ 2-category with only one object

Generalization:



A 0-signature

Σ_0

Example

signature

x y

POLYGRAPHS

A 0-rewriting system

$$\begin{tikzcd} & \Sigma_1 \arrow[dl, shift right, "s_0"'] \arrow[dl, shift left, "t_0"] \\ \Sigma_0 & \end{tikzcd}$$

Example

signature

$x \qquad y$

rules

$$\begin{tikzcd} x \arrow[r, "a"] & y \arrow[loop above, "b"] \end{tikzcd}$$

POLYGRAPHS

A 1-signature = a 0-rewriting system

$$\begin{tikzcd} & \Sigma_1 \arrow[dl, shift right, "s_0"'] \arrow[dl, shift left, "t_0"] \\ \Sigma_0 & \end{tikzcd}$$

Example

signature

$$\begin{tikzcd} x \arrow[r, "a"] & y \arrow[loop above, "b"] \end{tikzcd}$$

POLYGRAPHS

A 1-signature generates a *category*

$$
\begin{tikzcd}
 & \Sigma_1 \arrow[dl, shift right, "s_0"'] \arrow[dl, shift left, "t_0"] \arrow[d, "i_1"] \\
\Sigma_0 & \Sigma_1^* \arrow[l, shift right, "s_0^*"'] \arrow[l, shift left, "t_0^*"]
\end{tikzcd}
$$

Example

signature

$$
\begin{tikzcd}
x \arrow[r, "a"] & y \arrow[loop above, "b"]
\end{tikzcd}
$$

terms

$$
x \xrightarrow{a} y \xrightarrow{b} y \xrightarrow{b} y
$$

A 1-rewriting system

$$\begin{tikzcd}
& \Sigma_1 \arrow[dl, shift right, "s_0"'] \arrow[dl, shift left, "t_0"] \arrow[d, "i_1"] & \Sigma_2 \arrow[dl, shift right, "s_1"'] \arrow[dl, shift left, "t_1"] \\
\Sigma_0 & \Sigma_1^* \arrow[l, shift right, "s_0^*"'] \arrow[l, shift left, "t_0^*"] &
\end{tikzcd}$$

such that $s_0^* \circ s_1 = s_0^* \circ t_1$ and $t_0^* \circ s_1 = t_0^* \circ t_1$

Example

signature

$$\begin{tikzcd}
x \arrow[r, "a"] & y \arrow[loop above, "b"]
\end{tikzcd}$$

terms

$$x \xrightarrow{a} y \xrightarrow{b} y \xrightarrow{b} y$$

rules

$$\begin{tikzcd}
& y \arrow[r, "b"] & y \arrow[dr, "b"] & \\
x \arrow[ur, "a"] \arrow[rrr, bend right, "a"'] \arrow[rrr, phantom, "r\Downarrow"] & & & y
\end{tikzcd}$$

A 2-signature = a 1-rewriting system

$$
\begin{tikzcd}
 & \Sigma_1 \arrow[dl, shift right, "s_0"'] \arrow[dl, shift left, "t_0"] \arrow[d, "i_1"'] & \Sigma_2 \arrow[dl, shift right, "s_1"'] \arrow[dl, shift left, "t_1"] \\
\Sigma_0 & \Sigma_1^* \arrow[l, shift right, "s_0^*"'] \arrow[l, shift left, "t_0^*"] &
\end{tikzcd}
$$

such that $s_0^* \circ s_1 = s_0^* \circ t_1$ and $t_0^* \circ s_1 = t_0^* \circ t_1$

Example

signature

$$
\begin{tikzcd}
 & y \arrow[r, "b"] \arrow[d, phantom, "r\Downarrow"] & y \arrow[dr, "b"] \arrow[d, phantom, "\Downarrow s"] & \\
x \arrow[ur, "a"] \arrow[rrr, bend right, "a"'] & & & y
\end{tikzcd}
$$

POLYGRAPHS

A 2-signature generates a 2-category

$$
\begin{tikzcd}
 & \Sigma_1 \arrow[dl, shift right, "s_0"'] \arrow[dl, shift left, "t_0"] \arrow[d, "i_1"] & \Sigma_2 \arrow[dl, shift right, "s_1"'] \arrow[dl, shift left, "t_1"] \arrow[d, "i_2"] \\
\Sigma_0 & \Sigma_1^* \arrow[l, shift right, "s_0^*"'] \arrow[l, shift left, "t_0^*"] & \Sigma_2^* \arrow[l, shift right, "s_1^*"'] \arrow[l, shift left, "t_1^*"]
\end{tikzcd}
$$

such that $s_0^* \circ s_1 = s_0^* \circ t_1$ and $t_0^* \circ s_1 = t_0^* \circ t_1$

Example

signature

$$
\begin{tikzcd}
 & y \arrow[r, "b"] \arrow[d, phantom, "r\Downarrow"] & y \arrow[dr, "b"] \arrow[d, phantom, "\Downarrow s"] & \\
x \arrow[ur, "a"] \arrow[rrr, bend right, "a"'] & & & y
\end{tikzcd}
$$

POLYGRAPHS

A 2-rewriting system

$$\begin{tikzcd}
 & \Sigma_1 \arrow[dl, shift right, "s_0"'] \arrow[dl, shift left, "t_0"] \arrow[d, "i_1"] & \Sigma_2 \arrow[dl, shift right, "s_1"'] \arrow[dl, shift left, "t_1"] \arrow[d, "i_2"] & \Sigma_3 \arrow[dl, shift right, "s_2"'] \arrow[dl, shift left, "t_2"] \\
\Sigma_0 & \Sigma_1^* \arrow[l, shift right, "s_0^*"'] \arrow[l, shift left, "t_0^*"] & \Sigma_2^* \arrow[l, shift right, "s_1^*"'] \arrow[l, shift left, "t_1^*"] &
\end{tikzcd}$$

such that $s_1^* \circ s_2 = s_1^* \circ t_2$ and $t_1^* \circ s_2 = t_1^* \circ t_2$

Example

signature

$$\begin{tikzcd}
 & y \arrow[r, "b"] \arrow[d, phantom, "r\Downarrow"] & y \arrow[dr, "b"] \arrow[d, phantom, "\Downarrow s"] & \\
x \arrow[ur, "a"] \arrow[rrr, bend right, "a"'] & & & y
\end{tikzcd}$$

rules

$$\begin{tikzcd}
 & y \arrow[r, "b"] \arrow[dr, phantom, "r\Downarrow \overset{\alpha}{\Rrightarrow} \Downarrow s"] & y \arrow[dr, "b"] & \\
x \arrow[ur, "a"] \arrow[rrr, bend right, "a"'] & & & y
\end{tikzcd}$$

POLYGRAPHS

A 2-rewriting system

$$
\begin{tikzcd}
 & \Sigma_1 \arrow[dl, shift right, "s_0"'] \arrow[dl, shift left, "t_0"] \arrow[d, "i_1"'] & \Sigma_2 \arrow[dl, shift right, "s_1"'] \arrow[dl, shift left, "t_1"] \arrow[d, "i_2"'] & \Sigma_3 \arrow[dl, shift right, "s_2"'] \arrow[dl, shift left, "t_2"] \\
\Sigma_0 & \Sigma_1^* \arrow[l, shift right, "s_0^*"'] \arrow[l, shift left, "t_0^*"] & \Sigma_2^* \arrow[l, shift right, "s_1^*"'] \arrow[l, shift left, "t_1^*"] &
\end{tikzcd}
$$

Right notion of *n-rewriting system*: n-**polygraphs**.

`An example: a presentation of` **Bij**

[Lafont03]

A PRESENTATION OF Bij

The category **Bij** has

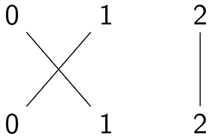
- objects are integers $[n] = \{0, \ldots, n-1\}$
- morphisms $f : [m] \to [n]$ are bijections

A PRESENTATION OF Bij

The category **Bij** has

- objects are integers $[n] = \{0, \ldots, n-1\}$
- morphisms $f : [m] \to [n]$ are bijections

$$\begin{tikzcd} {[3]} \arrow[d, "f"'] \\ {[3]} \end{tikzcd}$$

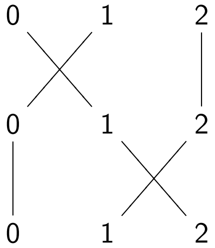


A PRESENTATION OF Bij

The category **Bij** has

- objects are integers $[n] = \{0, \ldots, n-1\}$
- morphisms $f : [m] \to [n]$ are bijections

$$
\begin{tikzcd}
{[3]} \arrow[d, "f"'] \\
{[3]} \arrow[d, "g"'] \\
{[3]}
\end{tikzcd}
$$

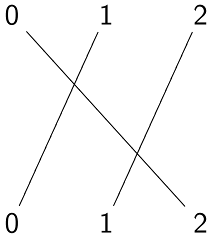


A PRESENTATION OF Bij

The category **Bij** has

- objects are integers $[n] = \{0, \ldots, n-1\}$
- morphisms $f : [m] \to [n]$ are bijections

$$\begin{tikzcd}[3] \arrow[d, "g \circ f"'] \\ [3]\end{tikzcd}$$



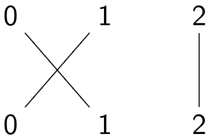
Vertical composition $\circ$

A PRESENTATION OF Bij

The category **Bij** has

- objects are integers $[n] = \{0, \ldots, n-1\}$
- morphisms $f : [m] \to [n]$ are bijections

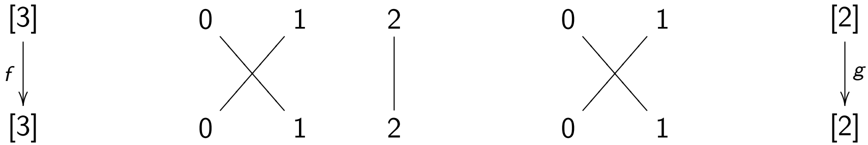
$$\begin{array}{c}[3]\\ \Big\downarrow{\scriptstyle f}\\ [3]\end{array}$$



A PRESENTATION OF Bij

The category **Bij** has

- objects are integers $[n] = \{0, \dots, n-1\}$
- morphisms $f : [m] \to [n]$ are bijections

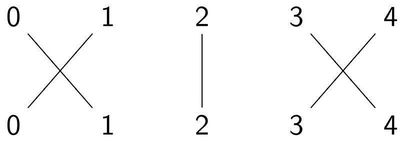


A PRESENTATION OF Bij

The category **Bij** has

- objects are integers $[n] = \{0, \ldots, n-1\}$
- morphisms $f : [m] \to [n]$ are bijections

$$\begin{tikzcd}
{[5]} \arrow[d, "f \otimes g"'] \\
{[5]}
\end{tikzcd}$$



Horizontal composition $\otimes$

A PRESENTATION OF Bij

We want to give a presentation of **Bij**, i.e. describe it as

- a free category on sets of typed generators for 0-, 1- and 2-cells
- quotiented by relations between 2-cells in the generated 2-category

A PRESENTATION OF Bij

Bij is presented by the 3-polygraph such that [Lafont03]

- $\Sigma_0 = \{*\}$

A PRESENTATION OF Bij

Bij is presented by the 3-polygraph such that [Lafont03]

- $\Sigma_0 = \{*\}$
- $\Sigma_1 = \{1 : * \to *\}$ (so $\Sigma_1^* \approx \mathbb{N}$)

A PRESENTATION OF Bij

Bij is presented by the 3-polygraph such that [Lafont03]

- $\Sigma_0 = \{*\}$
- $\Sigma_1 = \{1 : * \to *\}$ (so $\Sigma_1^* \approx \mathbb{N}$)
- $\Sigma_2 = \{\gamma : 1 \otimes 1 \to 1 \otimes 1\}$

$$\begin{tikzcd}
 & * \arrow[dr, "1"] & \\
* \arrow[ur, "1"] \arrow[dr, "1"'] \arrow[rr, phantom, "\gamma\Downarrow"] & & * \\
 & * \arrow[ur, "1"'] &
\end{tikzcd}$$

A PRESENTATION OF Bij

Bij is presented by the 3-polygraph such that [Lafont03]

- $\Sigma_0 = \{*\}$
- $\Sigma_1 = \{1 : * \to *\}$ (so $\Sigma_1^* \approx \mathbb{N}$)
- $\Sigma_2 = \{\gamma : 1 \otimes 1 \to 1 \otimes 1\}$
- $\Sigma_3 = \{y, s\}$

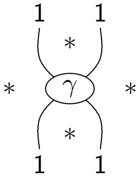
$$\begin{tikzcd} & * \arrow[dr, "1"] & \\ * \arrow[ur, "1"] \arrow[dr, "1"'] \arrow[rr, phantom, "\gamma\Downarrow"] & & * \\ & * \arrow[ur, "1"'] & \end{tikzcd}$$

$$\begin{tikzcd} & * \arrow[r] \arrow[d, phantom, "\gamma\Downarrow"] & * \arrow[dr] & \\ * \arrow[ur] \arrow[r] \arrow[dr] & * \arrow[ur] \arrow[dr] \arrow[d, phantom, "\gamma\Downarrow"] \arrow[rr, phantom, "\gamma\Downarrow"] & & * \\ & * \arrow[r] & * \arrow[ur] & \end{tikzcd} \overset{y}{\Rrightarrow} \begin{tikzcd} & * \arrow[r] \arrow[dr] & * \arrow[dr] \arrow[d, phantom, "\gamma\Downarrow"] & \\ * \arrow[ur] \arrow[dr] \arrow[rr, phantom, "\gamma\Downarrow"] & & * \arrow[r] \arrow[d, phantom, "\gamma\Downarrow"] & * \\ & * \arrow[ur] \arrow[r] & * \arrow[ur] & \end{tikzcd}$$

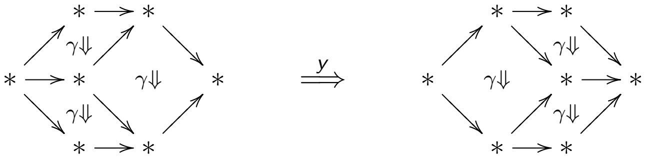
$$\begin{tikzcd} & * \arrow[dr] \arrow[d, phantom, "\gamma\Downarrow"] & \\ * \arrow[ur] \arrow[r] \arrow[dr] & * \arrow[r] \arrow[d, phantom, "\gamma\Downarrow"] & * \\ & * \arrow[ur] & \end{tikzcd} \overset{s}{\Rrightarrow} * \longrightarrow * \longrightarrow *$$

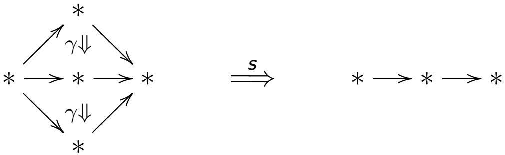
A PRESENTATION OF Bij

Bij is presented by the 3-polygraph such that [Lafont03]



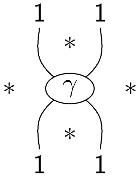
- $\Sigma_0 = \{*\}$
- $\Sigma_1 = \{1 : * \to *\}$ (so $\Sigma_1^* \approx \mathbb{N}$)
- $\Sigma_2 = \{\gamma : 1 \otimes 1 \to 1 \otimes 1\}$
- $\Sigma_3 = \{y, s\}$



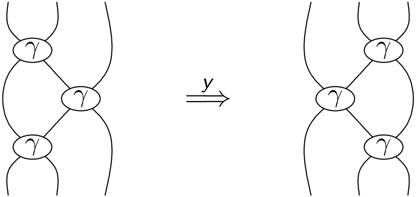


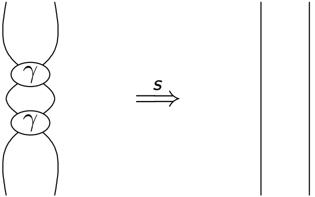
A PRESENTATION OF Bij

Bij is presented by the 3-polygraph such that [Lafont03]



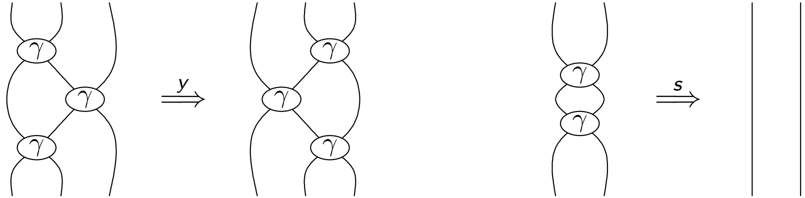
- $\Sigma_0 = \{*\}$
- $\Sigma_1 = \{1 : * \to *\}$ (so $\Sigma_1^* \approx \mathbb{N}$)
- $\Sigma_2 = \{\gamma : 1 \otimes 1 \to 1 \otimes 1\}$
- $\Sigma_3 = \{y, s\}$



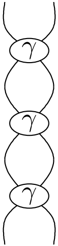


A PRESENTATION OF Bij

The rules

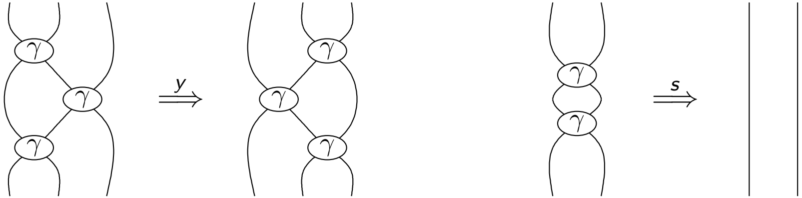


of the rewriting system induce critical pairs

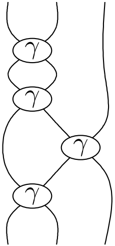


A PRESENTATION OF Bij

The rules

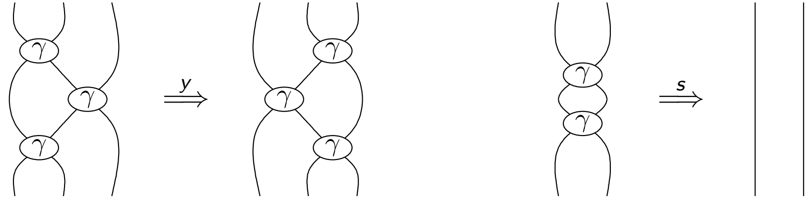


of the rewriting system induce critical pairs

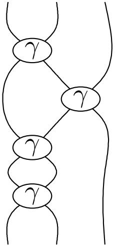


A PRESENTATION OF Bij

The rules

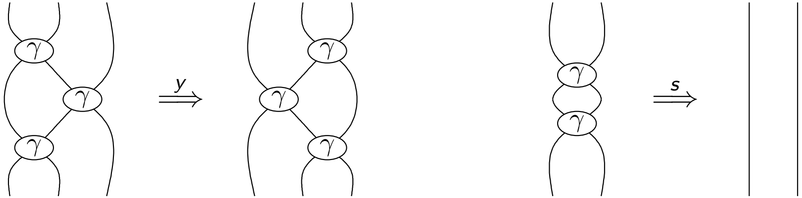


of the rewriting system induce critical pairs

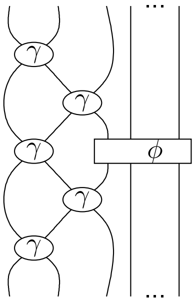


A PRESENTATION OF Bij

The rules



of the rewriting system induce critical pairs



Critical pairs are computed using a **unification** procedure.

We want to extend it to 2-dimensional rewriting systems

Contrarily to term rewriting systems
we can have an infinite number
of critical pairs...

IDEA:

change the definition

of critical pairs

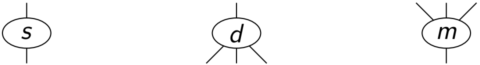
CRITICAL PAIRS
IN THE MULTICATEGORY OF
COMPACT CONTEXTS

TWO PROBLEMS WITH CRITICAL PAIRS

Consider the 2-rewriting system Σ with

$$\Sigma_0 = \{*\} \qquad \Sigma_1 = \{1\} \qquad \Sigma_2 = \{s : 1 \to 1,\ d : 1 \to 3,\ m : 3 \to 1\}$$

the generators for 2-cells are drawn respectively as

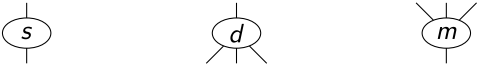


TWO PROBLEMS WITH CRITICAL PAIRS

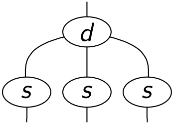
Consider the 2-rewriting system Σ with

$$\Sigma_0 = \{*\} \qquad \Sigma_1 = \{1\} \qquad \Sigma_2 = \{s : 1 \to 1,\ d : 1 \to 3,\ m : 3 \to 1\}$$

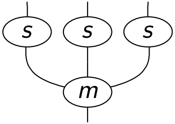
the generators for 2-cells are drawn respectively as



with rules



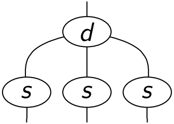
$\Rightarrow \dots$ and



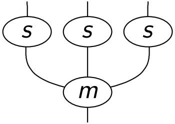
$\Rightarrow \dots$

TWO PROBLEMS WITH CRITICAL PAIRS

The two rules



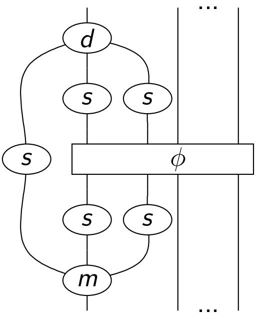
$\Rightarrow \ldots$ and



$\Rightarrow \ldots$

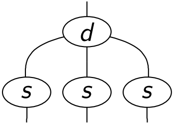
induce an infinite number of critical pairs:

variables on the border:

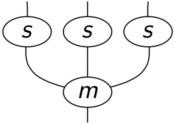


TWO PROBLEMS WITH CRITICAL PAIRS

The two rules



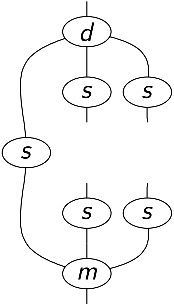
$\Rightarrow \dots$ and



$\Rightarrow \dots$

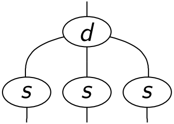
induce an infinite number of critical pairs:

variables on the border:
use compact morphisms!

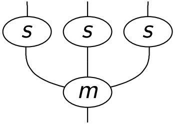


TWO PROBLEMS WITH CRITICAL PAIRS

The two rules



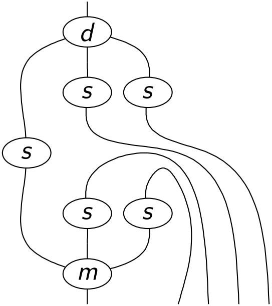
$\Rightarrow \dots$ and



$\Rightarrow \dots$

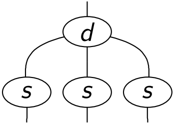
induce an infinite number of critical pairs:

variables on the border:
use compact morphisms!

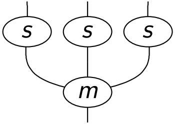


TWO PROBLEMS WITH CRITICAL PAIRS

The two rules



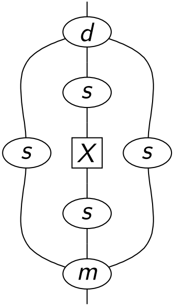
$\Rightarrow \ldots$ and



$\Rightarrow \ldots$

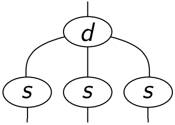
induce an infinite number of critical pairs:

variables inside:

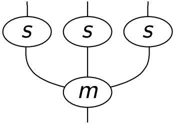


TWO PROBLEMS WITH CRITICAL PAIRS

The two rules



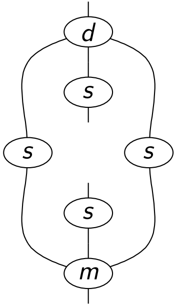
$\Rightarrow \ldots$ and



$\Rightarrow \ldots$

induce an infinite number of critical pairs:

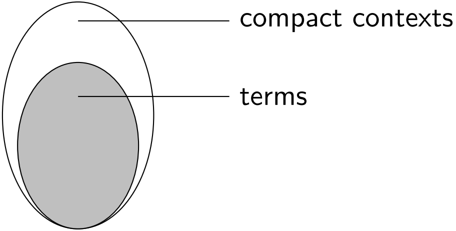
variables inside:
use contexts!
($=$ multicategory of
terms with metavariables)



BACK TO A FINITE NUMBER OF CRITICAL PAIRS

Theorem

The 2-category of "terms" generated by a signature can be embedded into the ***multicategory of compact contexts****.*



In other words, there is a finite number of *generating families* of critical pairs in those rewriting systems.

THE MULTICATEGORY OF CONTEXTS

Consider a 2-polygraph Σ

$$\begin{tikzcd}
 & \Sigma_1 \arrow[dl, shift right, "s_0"'] \arrow[dl, shift left, "t_0"] \arrow[d, "i_1"] & \Sigma_2 \arrow[dl, shift right, "s_1"'] \arrow[dl, shift left, "t_1"] \\
\Sigma_0 & \Sigma_1^* \arrow[l, shift right, "s_0^*"'] \arrow[l, shift left, "t_0^*"] &
\end{tikzcd}$$

and a family $\mathcal{X} = \{X_1 : f_1 \Rightarrow g_1, \ldots, X_n : f_n \Rightarrow g_n\}$ of 2-globes, with $f_i, g_i \in \Sigma_1^*$ parallel 1-cells considered as *formal variables*

THE MULTICATEGORY OF CONTEXTS

Consider a 2-polygraph Σ

$$\begin{tikzcd}
 & \Sigma_1 \arrow[dl, shift right, "s_0"'] \arrow[dl, shift left, "t_0"] \arrow[d, "i_1"] & \Sigma_2 \arrow[dl, shift right, "s_1"'] \arrow[dl, shift left, "t_1"] \\
\Sigma_0 & \Sigma_1^* \arrow[l, shift right, "s_0^*"'] \arrow[l, shift left, "t_0^*"] &
\end{tikzcd}$$

and a family $\mathcal{X} = \{X_1 : f_1 \Rightarrow g_1, \ldots, X_n : f_n \Rightarrow g_n\}$ of 2-globes, with $f_i, g_i \in \Sigma_1^*$ parallel 1-cells considered as *formal variables*

We define the 2-polygraph $\Sigma[\mathcal{X}]$ as

$$\begin{tikzcd}
 & \Sigma_1 \arrow[dl, shift right, "s_0"'] \arrow[dl, shift left, "t_0"] \arrow[d, "i_1"] & \Sigma_2 \uplus \mathcal{X} \arrow[dl, shift right, "s_1"'] \arrow[dl, shift left, "t_1"] \\
\Sigma_0 & \Sigma_1^* \arrow[l, shift right, "s_0^*"'] \arrow[l, shift left, "t_0^*"] &
\end{tikzcd}$$

THE MULTICATEGORY OF CONTEXTS

Consider a 2-polygraph Σ

$$
\begin{tikzcd}
 & \Sigma_1 \arrow[dl, shift right, "s_0"'] \arrow[dl, shift left, "t_0"] \arrow[d, "i_1"] & \Sigma_2 \arrow[dl, shift right, "s_1"'] \arrow[dl, shift left, "t_1"] \arrow[d, "i_2"] \\
\Sigma_0 & \Sigma_1^* \arrow[l, shift right, "s_0^*"'] \arrow[l, shift left, "t_0^*"] & \Sigma_2^* \arrow[l, shift right, "s_1^*"'] \arrow[l, shift left, "t_1^*"]
\end{tikzcd}
$$

and a family $\mathcal{X} = \{X_1 : f_1 \Rightarrow g_1, \ldots, X_n : f_n \Rightarrow g_n\}$ of 2-globes, with $f_i, g_i \in \Sigma_1^*$ parallel 1-cells considered as *formal variables*

We define the 2-polygraph $\Sigma[\mathcal{X}]$ as

$$
\begin{tikzcd}
 & \Sigma_1 \arrow[dl, shift right, "s_0"'] \arrow[dl, shift left, "t_0"] \arrow[d, "i_1"] & \Sigma_2 \uplus \mathcal{X} \arrow[dl, shift right, "s_1"'] \arrow[dl, shift left, "t_1"] \arrow[d, "i_2"] \\
\Sigma_0 & \Sigma_1^* \arrow[l, shift right, "s_0^*"'] \arrow[l, shift left, "t_0^*"] & (\Sigma_2 \uplus \mathcal{X})^* \arrow[l, shift right, "s_1^*"'] \arrow[l, shift left, "t_1^*"]
\end{tikzcd}
$$

THE MULTICATEGORY OF CONTEXTS

Consider a 2-polygraph Σ

$$\begin{tikzcd}
 & \Sigma_1 \arrow[dl, shift right, "s_0"'] \arrow[dl, shift left, "t_0"] \arrow[d, "i_1"'] & \Sigma_2 \arrow[dl, shift right, "s_1"'] \arrow[dl, shift left, "t_1"] \arrow[d, "i_2"'] \\
\Sigma_0 & \Sigma_1^* \arrow[l, shift right, "s_0^*"'] \arrow[l, shift left, "t_0^*"] & \Sigma_2^* \arrow[l, shift right, "s_1^*"'] \arrow[l, shift left, "t_1^*"]
\end{tikzcd}$$

Substitution

Given

- a 2-cell $\alpha : f \Rightarrow f'$ in Σ_2^*
- and a 2-cell $\beta : g \Rightarrow g'$ in $\Sigma_2[X : f \Rightarrow f']^*$

we can define

- a 2-cell $\alpha[\beta/X] : f \Rightarrow f'$ in Σ_2^*

which corresponds to the 2-cell α where all occurrences of X have been replaced by β.

THE MULTICATEGORY OF CONTEXTS

Consider a 2-polygraph Σ

$$\begin{tikzcd}
 & \Sigma_1 \arrow[dl, shift right, "s_0"'] \arrow[dl, shift left, "t_0"] \arrow[d, "i_1"] & \Sigma_2 \arrow[dl, shift right, "s_1"'] \arrow[dl, shift left, "t_1"] \arrow[d, "i_2"] \\
\Sigma_0 & \Sigma_1^* \arrow[l, shift right, "s_0^*"'] \arrow[l, shift left, "t_0^*"] & \Sigma_2^* \arrow[l, shift right, "s_1^*"'] \arrow[l, shift left, "t_1^*"]
\end{tikzcd}$$

Definition

We can thus define the **multicategory of contexts** $\mathcal{K}_\Sigma$ whose

- objects are globes $f \Rightarrow g$, i.e. parallel 1-cells in Σ_1^*
- operations in $\mathcal{K}_\Sigma(f_1 \Rightarrow g_1, \ldots, f_n \Rightarrow g_n; f \Rightarrow g)$ are 2-cells $\kappa : f \Rightarrow g$ in $\Sigma_2[X_1 : f_1 \Rightarrow g_1, \ldots, X_n : f_n \Rightarrow g_n]^*$
- composition is given by generalizing the previous substitution

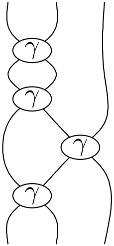
CRITICAL PAIRS

Definition

Suppose that we are given two 2-cells

$$\alpha_1 : f_1 \Rightarrow g_1 \qquad \text{and} \qquad \beta_1 : f_1 \Rightarrow g_1$$

in a 2-polygraph Σ. A **most general unifier** is



CRITICAL PAIRS

Definition

Suppose that we are given two 2-cells

$$\alpha_1 : f_1 \Rightarrow g_1 \qquad \text{and} \qquad \beta_1 : f_1 \Rightarrow g_1$$

in a 2-polygraph Σ. A **most general unifier** is a pair

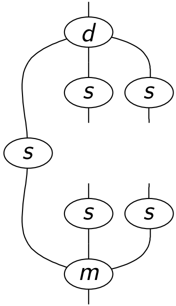
$$\kappa_1 \in \mathcal{K}_\Sigma(f_1 \Rightarrow g_1; f \Rightarrow g) \qquad \text{and} \qquad \kappa_2 \in \mathcal{K}_\Sigma(f_2 \Rightarrow g_2; f \Rightarrow g)$$

of *linear* contexts such that

1. *unifier*: $\kappa_1(\alpha_1) = \kappa_2(\alpha_2)$
2. *minimal*: if $\kappa_1 = \kappa_1'' \circ \kappa_1'$ and $\kappa_2 = \kappa_2'' \circ \kappa_2'$ where (κ_1', κ_2') is a unifier then $\kappa_1'' = \mathrm{id}$ and $\kappa_2'' = \mathrm{id}$
3. *overlapping*: there is no binary context κ such that $\kappa_1 = (\mathrm{id}, \alpha_2)$ and $\kappa_2 = (\alpha_1, \mathrm{id})$

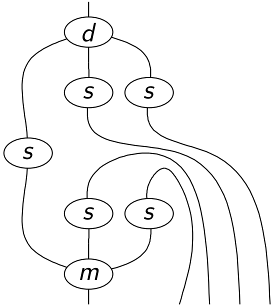
COMPACT 2-CATEGORIES

Now, we want to represent morphisms with "holes" in the border



COMPACT 2-CATEGORIES

Now, we want to represent morphisms with "holes" in the border



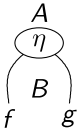
COMPACT 2-CATEGORIES

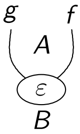
Definition

Given a 2-category $\mathcal{C}$, a 1-cell $f : A \to B$ is **left adjoint** to a 1-cell $g : B \to A$, which we write

$$\begin{tikzcd} A \arrow[r, bend left, "f"] \arrow[r, phantom, "\perp"] & B \arrow[l, bend left, "g"] \end{tikzcd}$$

when there exists two 2-cells $\eta : A \to f \otimes g$ and $\varepsilon : g \otimes f \to B$





such that

$$(f \otimes \varepsilon) \circ (\eta \otimes f) = f \quad \text{and} \quad (\varepsilon \otimes g) \circ (g \otimes \eta) = g$$

COMPACT 2-CATEGORIES

Definition

A 2-category $\mathcal{C}$ is **compact** when every 1-cell admits both a left and a right adjoint.

COMPACT 2-CATEGORIES

Definition

A 2-category $\mathcal{C}$ is **compact** when every 1-cell admits both a left and a right adjoint.

Lemma

A 2-category $\mathcal{C}$ generates a free compact 2-category $\mathcal{A}_\mathcal{C}$, which is

- 0-*cells: same as* $\mathcal{C}$,
- 1-*cells:* f^n *with* f *a 1-cell of* $\mathcal{C}$ *and* $n \in \mathbb{Z}$,
- 2-*cells:*
  - $\alpha : f^0 \Rightarrow g^0$ *for* $\alpha : f \Rightarrow g$ *a 2-cell of* $\mathcal{C}$,
  - $\eta_{f^n} : \mathrm{id} \Rightarrow f^{n-1} \otimes f^n$
  - $\varepsilon_{f^n} : f^n \otimes f^{n-1} \Rightarrow \mathrm{id}$
- $+$ *equations*

COMPACT 2-CATEGORIES

Definition

A 2-category $\mathcal{C}$ is **compact** when every 1-cell admits both a left and a right adjoint.

Lemma

A 2-category $\mathcal{C}$ generates a free compact 2-category $\mathcal{A}_{\mathcal{C}}$, which is

- *0-cells: same as $\mathcal{C}$,*
- *1-cells: f^n with f a 1-cell of $\mathcal{C}$ and $n \in \mathbb{Z}$,*
- *2-cells:*
  - $\alpha : f^0 \Rightarrow g^0$ *for* $\alpha : f \Rightarrow g$ *a 2-cell of* $\mathcal{C}$,
  - $\eta_{f^n} : \mathrm{id} \Rightarrow f^{n-1} \otimes f^n$
  - $\varepsilon_{f^n} : f^n \otimes f^{n-1} \Rightarrow \mathrm{id}$
- $+$ *equations*

Theorem

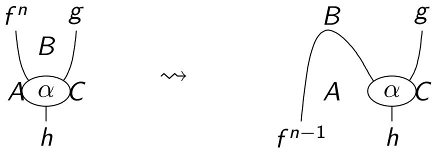
A 2-category $\mathcal{C}$ embeds fully and faithfully in the free compact category it generates.

ROTATIONS

In the free compact 2-category $\mathcal{A}_{\mathcal{C}}$, the following Hom-sets

$$\mathcal{A}_{\mathcal{C}}(f^n \otimes g, h) \qquad \cong \qquad \mathcal{A}_{\mathcal{C}}(g, f^{n-1} \otimes h)$$

are isomorphic:



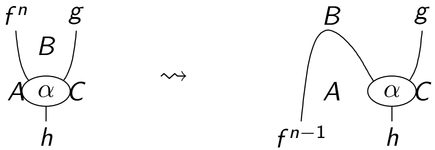
We call these isomorphisms **rotations**.

ROTATIONS

In the free compact 2-category $\mathcal{A}_\mathcal{C}$, the following Hom-sets

$$\mathcal{A}_\mathcal{C}(f^n \otimes g, h) \qquad \cong \qquad \mathcal{A}_\mathcal{C}(g, f^{n-1} \otimes h)$$

are isomorphic:

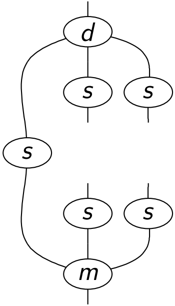


We call these isomorphisms **rotations**.

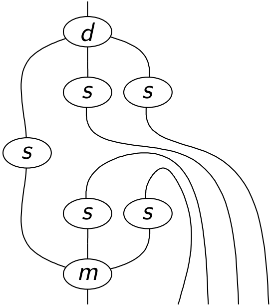
Rotations are unary operations in $\mathcal{K}_{\mathcal{A}_\mathcal{C}}$.

ROTATIONS

The diagram



is a representation of



up to rotation!

THE MULTICATEGORY OF COMPACT CONTEXTS

A 2-polygraph Σ generates a 2-category $\mathcal{C}$,

- which can be embedded in the free compact 2-category $\mathcal{A}_{\mathcal{C}}$
- and the 2-cells $\alpha : f \Rightarrow g$ of $\mathcal{A}_{\mathcal{C}}$ can be seen as nullary contexts in $\mathcal{K}_{\mathcal{A}_{\mathcal{C}}}(; f \Rightarrow g)$

UNIFICATION IN THE MULTICATEGORY OF COMPACT CONTEXTS

Theorem

Given a 3-polygraph R with underlying 2-polygraph Σ generating a 2-category $\mathcal{C}$, there exists a ***finite*** *number of contexts*

$$\kappa^i \quad \in \quad \mathcal{K}_{\mathcal{A}_\mathcal{C}}(f_1^i \Rightarrow g_1^i \ , \ \ldots \ , \ f_{k_i}^i \Rightarrow g_{k_i}^i ; f^i \Rightarrow g^i)$$

such that

- *for any nullary contexts $\kappa_1, \ldots, \kappa_{k_i}$ and unary context κ such that the composite $\kappa \circ \kappa^i \circ (\kappa_1, \ldots, \kappa_{k_i}) \in \mathcal{K}_{\mathcal{A}_\mathcal{C}}(; f' \Rightarrow g')$ is of the form κ_α, for some 2-cell $\alpha : f' \Rightarrow g'$ of $\mathcal{C}$, the 2-cell α is an unifier of left members of two rewriting rules of R,*
- *and moreover any such unifier can be obtained in this way (in particular the critical pairs).*

UNIFICATION IN THE MULTICATEGORY OF COMPACT CONTEXTS

Theorem

*Given a 3-polygraph R with underlying 2-polygraph Σ generating a 2-category $\mathcal{C}$, there exists a **finite** number of contexts*

$$\kappa^i \quad \in \quad \mathcal{K}_{\mathcal{A}_\mathcal{C}}(f_1^i \Rightarrow g_1^i \ , \ \ldots \ , \ f_{k_i}^i \Rightarrow g_{k_i}^i ; f^i \Rightarrow g^i)$$

such that

- *for any nullary contexts $\kappa_1, \ldots, \kappa_{k_i}$ and unary context κ such that the composite $\kappa \circ \kappa^i \circ (\kappa_1, \ldots, \kappa_{k_i}) \in \mathcal{K}_{\mathcal{A}_\mathcal{C}}(; f' \Rightarrow g')$ is of the form κ_α, for some 2-cell $\alpha : f' \Rightarrow g'$ of $\mathcal{C}$, the 2-cell α is an unifier of left members of two rewriting rules of R,*
- *and moreover any such unifier can be obtained in this way (in particular the critical pairs).*

Remark

There is unicity up to rotations.

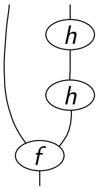
UNIFICATION
IN THE MULTICATEGORY OF
COMPACT CONTEXTS

UNIFICATION IN TRS

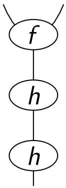
Suppose that we have a TRS

$f : 2 \qquad h : 1 \qquad\qquad f(x, h(h(y))) \Rightarrow \dots \qquad h(h(f(x, y))) \Rightarrow \dots$

In order to generate critical pairs,
we unify a subterm of $f(x, h(h(y)))$ with $h(h(f(x, y)))$



$\stackrel{?}{=}$

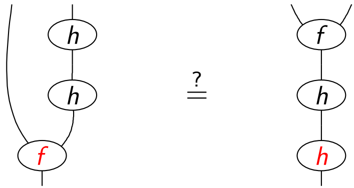


UNIFICATION IN TRS

Suppose that we have a TRS

$f : 2 \qquad h : 1 \qquad\qquad f(x, h(h(y))) \Rightarrow \ldots \qquad h(h(f(x, y))) \Rightarrow \ldots$

In order to generate critical pairs,
we unify a subterm of $f(x, h(h(y)))$ with $h(h(f(x, y)))$

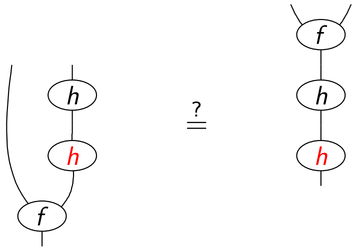


UNIFICATION IN TRS

Suppose that we have a TRS

$f : 2 \qquad h : 1 \qquad\qquad f(x, h(h(y))) \Rightarrow \ldots \qquad h(h(f(x, y))) \Rightarrow \ldots$

In order to generate critical pairs,
we unify a subterm of $f(x, h(h(y)))$ with $h(h(f(x, y)))$

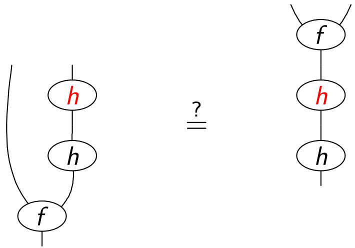


UNIFICATION IN TRS

Suppose that we have a TRS

$f : 2 \qquad h : 1 \qquad\qquad f(x, h(h(y))) \Rightarrow \ldots \qquad h(h(f(x, y))) \Rightarrow \ldots$

In order to generate critical pairs,
we unify a subterm of $f(x, h(h(y)))$ with $h(h(f(x, y)))$

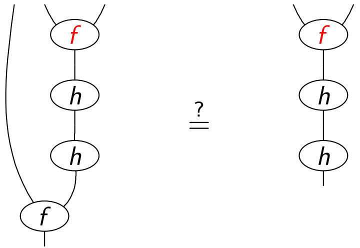


UNIFICATION IN TRS

Suppose that we have a TRS

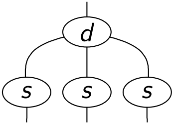
$f : 2 \qquad h : 1 \qquad\qquad f(x, h(h(y))) \Rightarrow \ldots \qquad h(h(f(x, y))) \Rightarrow \ldots$

In order to generate critical pairs,
we unify a subterm of $f(x, h(h(y)))$ with $h(h(f(x, y)))$

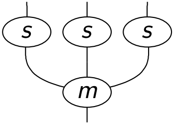


UNIFICATION IN 2-REWRITING SYSTEMS

We want to compute the critical pairs generated by

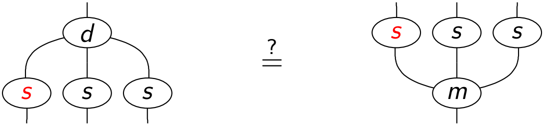


$\overset{?}{=}$



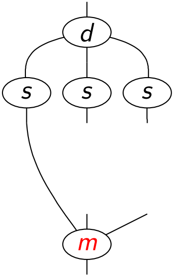
UNIFICATION IN 2-REWRITING SYSTEMS

We want to compute the critical pairs generated by

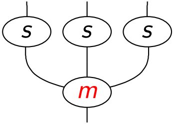


UNIFICATION IN 2-REWRITING SYSTEMS

We want to compute the critical pairs generated by

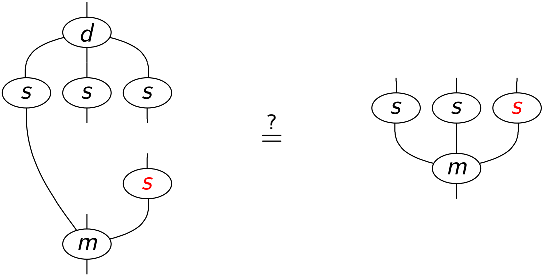


$\stackrel{?}{=}$



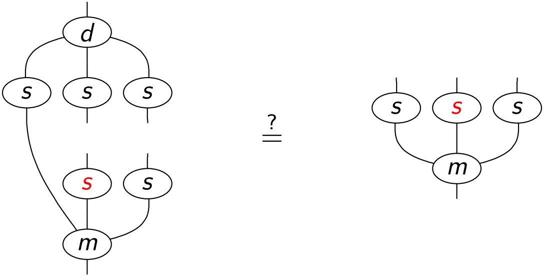
UNIFICATION IN 2-REWRITING SYSTEMS

We want to compute the critical pairs generated by



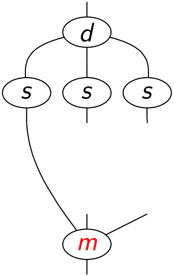
UNIFICATION IN 2-REWRITING SYSTEMS

We want to compute the critical pairs generated by

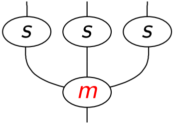


UNIFICATION IN 2-REWRITING SYSTEMS

We want to compute the critical pairs generated by

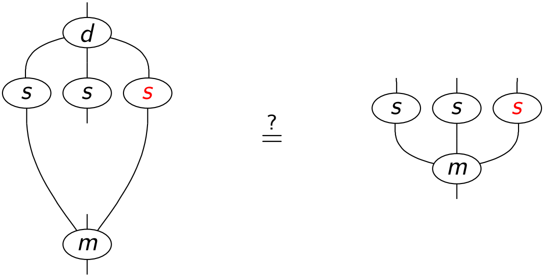


$$\stackrel{?}{=}$$



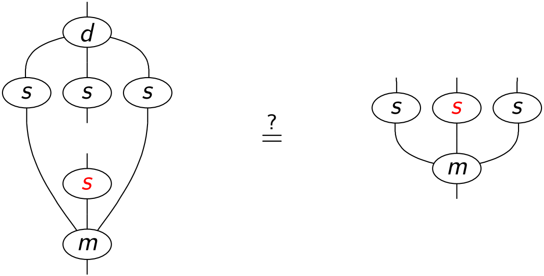
UNIFICATION IN 2-REWRITING SYSTEMS

We want to compute the critical pairs generated by



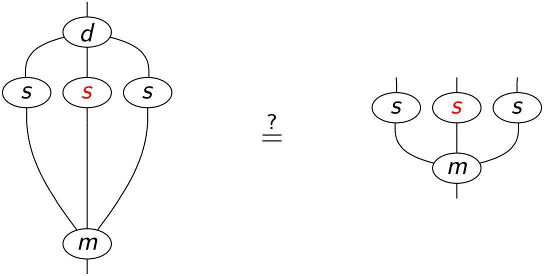
UNIFICATION IN 2-REWRITING SYSTEMS

We want to compute the critical pairs generated by



UNIFICATION IN 2-REWRITING SYSTEMS

We want to compute the critical pairs generated by



METACONFLUENCE

Remark

When a 2-rewriting system is confluent, its critical pairs (in our generalized sense) are not necessarily confluent. But at least, we get a finite description of the critical pairs!

NOT SHOWN HERE

- The operations in the multicategory of compact contexts can be represented in an effective way.

NOT SHOWN HERE

- The operations in the multicategory of compact contexts can be represented in an effective way.
- Precise formulation of the algorithm.

NOT SHOWN HERE

- The operations in the multicategory of compact contexts can be represented in an effective way.
- Precise formulation of the algorithm.
- An implementation was realized.

FUTURE WORKS

- Generalize techniques developed by Guiraud and Malbos to this setting
- Generalize to higher dimensions
- Towards automated tools for studying higher categories?
- ...

THANKS!

Any questions?