HIGHER-DIMENSIONAL REWRITING SYSTEMS

SAMUEL MIMRAM

IWC/TERMGRAPH invited talk

JULY 13th 2014

A panorama

1. *Motivation*: presentations of monoids and categories
2. *Definition*: higher-dimensional rewriting systems
3. *Example*: the theory for monoids
4. *Application*: coherence for monoidal categories
5. *Technique*: proving confluence
6. *Sanity check*: encoding term rewriting systems
7. *Conclusion*

UNIFYING REWRITING FORMALISMS

A unifying framework

There are many flavors of rewriting systems:

- abstract rewriting systems
- string rewriting systems
- term rewriting systems
- graph rewriting systems
- …

Do they fit in some general pattern?

The idea of dimension

The idea of dimension appears in many contexts:

- in topology: a manifold of dimension n is a topological space which locally looks like $\mathbb{R}^n$
- in algebraic topology: one consider points, paths, homotopies between paths, homotopies between homotopies, etc.
- in category theory: an n-category consists of collections of k-cells for $0 \leq k \leq n$

$$\begin{tikzcd} A \arrow[r, bend left, "f"] \arrow[r, bend right, "g"'] \arrow[r, phantom, "\Downarrow \alpha"] & B \end{tikzcd}$$

What is a rewriting system of dimension n?

Higher-dimensional rewriting systems

I will present higher-dimensional rewriting systems, fitting the following specification:

string rewriting system
=
presentation of a monoid

Higher-dimensional rewriting systems

I will present higher-dimensional rewriting systems, fitting the following specification:

$(n+1)$-dimensional rewriting system

=

presentation of an n-category

Higher-dimensional rewriting systems

I will present higher-dimensional rewriting systems, fitting the following specification:

$(n+1)$-dimensional rewriting system
$=$
presentation of an n-category

We will see that

- previous cases will be recovered as particular cases,

Higher-dimensional rewriting systems

I will present higher-dimensional rewriting systems, fitting the following specification:

$(n+1)$-dimensional rewriting system
=
presentation of an n-category

We will see that

- previous cases will be recovered as particular cases,
- we get a nice inductive definition:

an $(n+1)$-rewriting system is given by rules
rewriting
rewriting paths in an n-dimensional rewriting system

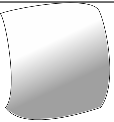
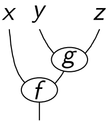
The intuition of dimension

	Geometry	Rewriting systems
0	$\cdot$	$\cdot x$

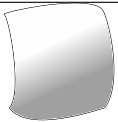
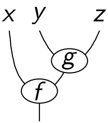
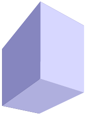
The intuition of dimension

	Geometry	Rewriting systems
0	$\cdot$	$\cdot x$
1		$\xrightarrow{a} \xrightarrow{b} \xrightarrow{c}$

The intuition of dimension

	Geometry	Rewriting systems
0	$\cdot$	$\cdot x$
1		$\xrightarrow{a} \xrightarrow{b} \xrightarrow{c}$
2		

The intuition of dimension

	Geometry	Rewriting systems
0	$\cdot$	$\cdot x$
1		$\xrightarrow{a}\ \xrightarrow{b}\ \xrightarrow{c}$
2		
3		???
$\dots$	$\dots$	???

Historical perspective

Most of the work I will present here has been done by others.

- 1976: **2-computads**
  R. Street, *Limits indexed by category-valued 2-functors*
- 1990: ***n*-computads**
  J. Power, *An n-Categorical Pasting Theorem*
- 1993: ***n*-polygraphs**
  A. Burroni, *Higher dimensional word problems with applications to equational logic*
- 2003: critical pairs in 3-dimensional rs
  Y. Lafont, *Towards an Algebraic Theory of Boolean Circuits*
- many other people: Y. Guiraud, P. Malbos, F. Métayer, …

Abstract rewriting systems

An **abstract rewriting system** consists of

- a set Σ_0 of *terms*
- a set $\Sigma_1 \subseteq \Sigma_0 \times \Sigma_0$ of *rules*

Abstract rewriting systems

An **abstract rewriting system** consists of

- a set Σ_0 of *terms*
- a set Σ_1 of *rules* together with two functions $s_0, t_0 : \Sigma_1 \to \Sigma_0$

Abstract rewriting systems

An **abstract rewriting system** consists of

- a set Σ_0 of *terms*
- a set Σ_1 of *rules* together with two functions $s_0, t_0 : \Sigma_1 \to \Sigma_0$

We write

$$r \quad : \quad x \quad \to \quad y$$

whenever $r \in \Sigma_0$ with $s(r) = x$ and $t(r) = y$.

Abstract rewriting systems

An **abstract rewriting system** consists of

- a set Σ_0 of *terms*
- a set Σ_1 of *rules* together with two functions $s_0, t_0 : \Sigma_1 \to \Sigma_0$

We write

$$r \quad : \quad x \quad \to \quad y$$

whenever $r \in \Sigma_0$ with $s(r) = x$ and $t(r) = y$.

When the rewriting system is terminating and confluent, normal forms are in bijection with equivalence classes in

$$\Sigma_0/\overset{*}{\leftrightarrow}$$

Abstract rewriting systems

An **abstract rewriting system** consists of

- a set Σ_0 of *terms*
- a set Σ_1 of *rules* together with two functions $s_0, t_0 : \Sigma_1 \to \Sigma_0$

We write

$$r \quad : \quad x \quad \to \quad y$$

whenever $r \in \Sigma_0$ with $s(r) = x$ and $t(r) = y$.

Notice that an abstract rewriting system is simply a graph!

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

1-dimensional rewriting systems

A **1-dimensional rewriting system** is an ARS, i.e. a graph

$$\Sigma_0 \underset{t_0}{\overset{s_0}{\leftleftarrows}} \Sigma_1$$

1-dimensional rewriting systems

A **1-dimensional rewriting system** is an ARS, i.e. a graph

$$\Sigma_0 \underset{t_0}{\overset{s_0}{\leftleftarrows}} \Sigma_1$$

We say that it **presents** the set

$$\Sigma_0 / \overset{*}{\leftrightarrow}$$

What about string rewriting systems?

String rewriting systems

A **string rewriting system** consists of

- a set Σ of *letters*
- a set $R \subseteq \Sigma^* \times \Sigma^*$ of *rules*

String rewriting systems

A **string rewriting system** consists of

- a set Σ of *letters*
- a set R of *rules* together with $s, t : R \to \Sigma^*$

A **string rewriting system** consists of

- a set Σ of *letters*
- a set R of *rules* together with $s, t : R \to \Sigma^*$

Given a rule

$$r \quad : \quad v \quad \to \quad v'$$

a *rewriting step* is of the form

$$urw \quad : \quad uvw \quad \to \quad uv'w$$

with $u, w \in \Sigma^*$.

The relation $\overset{*}{\leftrightarrow}$ is a congruence wrt concatenation:

$$u \overset{*}{\leftrightarrow} u' \quad \text{and} \quad v \overset{*}{\leftrightarrow} v' \qquad \text{implies} \qquad uv \overset{*}{\leftrightarrow} u'v'$$

The relation $\overset{*}{\leftrightarrow}$ is a congruence wrt concatenation:

$$u \overset{*}{\leftrightarrow} u' \quad \text{and} \quad v \overset{*}{\leftrightarrow} v' \qquad \text{implies} \qquad uv \overset{*}{\leftrightarrow} u'v'$$

we thus have a quotient *monoid*

$$\Sigma^*/\overset{*}{\leftrightarrow}$$

The relation $\overset{*}{\leftrightarrow}$ is a congruence wrt concatenation:

$$u \overset{*}{\leftrightarrow} u' \quad \text{and} \quad v \overset{*}{\leftrightarrow} v' \qquad \text{implies} \qquad uv \overset{*}{\leftrightarrow} u'v'$$

we thus have a quotient *monoid*

$$\Sigma^*/\overset{*}{\leftrightarrow}$$

We say that a rewriting system (Σ, R) is a **presentation** of a monoid M whenever

$$M \quad \cong \quad \Sigma^*/\overset{*}{\leftrightarrow}$$

String rewriting systems

The relation $\overset{*}{\leftrightarrow}$ is a congruence wrt concatenation:

$$u \overset{*}{\leftrightarrow} u' \quad \text{and} \quad v \overset{*}{\leftrightarrow} v' \qquad \text{implies} \qquad uv \overset{*}{\leftrightarrow} u'v'$$

we thus have a quotient *monoid*

$$\Sigma^* / \overset{*}{\leftrightarrow}$$

We say that a rewriting system (Σ, R) is a **presentation** of a monoid M whenever

$$M \quad \cong \quad \Sigma^* / \overset{*}{\leftrightarrow}$$

When the rewriting system is terminating and confluent, this means that elements of M are in bijection with normal forms in Σ^*.

For instance, consider the additive monoid

$$\mathbb{N} \times (\mathbb{N}/2\mathbb{N})$$

For instance, consider the additive monoid

$$\mathbb{N} \times (\mathbb{N}/2\mathbb{N})$$

I claim that a presentation for this monoid is given by

$$\Sigma = \{a, b\} \qquad R = \{ba \to ab, bb \to \varepsilon\}$$

Building presentations

For instance, consider the additive monoid

$$\mathbb{N} \times (\mathbb{N}/2\mathbb{N})$$

I claim that a presentation for this monoid is given by

$$\Sigma = \{a, b\} \qquad R = \{ba \to ab, bb \to \varepsilon\}$$

Critical pairs are:

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

For instance, consider the additive monoid

$$\mathbb{N} \times (\mathbb{N}/2\mathbb{N})$$

I claim that a presentation for this monoid is given by

$$\Sigma = \{a, b\} \qquad R = \{ba \to ab, bb \to \varepsilon\}$$

Critical pairs are joinable:

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

For instance, consider the additive monoid

$$\mathbb{N} \times (\mathbb{N}/2\mathbb{N})$$

I claim that a presentation for this monoid is given by

$$\Sigma = \{a, b\} \qquad R = \{ba \to ab, bb \to \varepsilon\}$$

The rewriting system is terminating…

Building presentations

For instance, consider the additive monoid

$$\mathbb{N} \times (\mathbb{N}/2\mathbb{N})$$

I claim that a presentation for this monoid is given by

$$\Sigma = \{a, b\} \qquad R = \{ba \to ab, bb \to \varepsilon\}$$

Normal forms are:

$$a^n \qquad \text{and} \qquad a^n b$$

and are in bijection with elements of $\mathbb{N} \times (\mathbb{N}/2\mathbb{N})$:

$$\mathbb{N} \times (\mathbb{N}/2\mathbb{N}) \quad \cong \quad \Sigma^* / \overset{*}{\leftrightarrow}_R$$

Presentations of monoids

String rewriting systems are useful in order to build presentations of monoids, from which

- we get a small (e.g. finite) description of the monoid
- we can perform computations (e.g. homology, etc.)

Presentations of monoids

String rewriting systems are useful in order to build presentations of monoids, from which

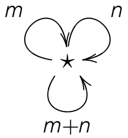
- we get a small (e.g. finite) description of the monoid
- we can perform computations (e.g. homology, etc.)

There are many other examples…

$$\mathfrak{S}_n \cong \langle \sigma_1, \ldots, \sigma_n \mid \sigma_i\sigma_{i+1}\sigma_i = \sigma_{i+1}\sigma_i\sigma_{i+1},\ \sigma_i^2 = 1,\ \sigma_i\sigma_j = \sigma_j\sigma_i \rangle$$

Presentations of categories

A monoid can be seen as a particular case of a category with only one object.



How do we modify rewriting systems
in order to
present categories instead of monoids?

A **2-dimensional rewriting system** is a pair (Σ, Σ_2) where

Presentations of categories

A **2-dimensional rewriting system** is a pair (Σ, Σ_2) where

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

- the "alphabet" Σ is now given by a graph

A **2-dimensional rewriting system** is a pair (Σ, Σ_2) where

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

- the "alphabet" Σ is now given by a graph and Σ^* denotes the free category on the graph:
  - objects are Σ_0: the objects of the graph
  - morphisms are Σ_1^*: the paths in the graph

A **2-dimensional rewriting system** is a pair (Σ, Σ_2) where

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

- the “alphabet” Σ is now given by a graph and Σ^* denotes the free category on the graph:
  - objects are Σ_0: the objects of the graph
  - morphisms are Σ_1^*: the paths in the graph
- the rules in Σ_2 are pairs of paths with same source and target

$$s_0^* \circ s_1 = s_0^* \circ t_1 \qquad t_0^* \circ s_1 = t_0^* \circ t_1$$

Recasting previous example

Consider the category with

- one object: $\star$
- $\mathbb{N} \times (\mathbb{N}/2\mathbb{N})$ as objects with addition as composition.

Recasting previous example

Consider the category with

- one object: $\star$
- $\mathbb{N} \times (\mathbb{N}/2\mathbb{N})$ as objects with addition as composition.

It admits the presentation with

- signature is the graph

$$\begin{tikzcd} \star \arrow[loop left, "a"] \arrow[loop right, "b"] \end{tikzcd}$$

Recasting previous example

Consider the category with

- one object: $\star$
- $\mathbb{N} \times (\mathbb{N}/2\mathbb{N})$ as objects with addition as composition.

It admits the presentation with

- signature is the graph

$$
\begin{tikzcd}
\star \arrow[loop left, "a"] \arrow[loop right, "b"]
\end{tikzcd}
$$

- rules are

$$
\begin{tikzcd}
& \star \arrow[dr, "a"] & \\
\star \arrow[ur, "b"] \arrow[dr, "a"'] \arrow[rr, phantom, "\Downarrow \alpha"] & & \star \\
& \star \arrow[ur, "b"'] &
\end{tikzcd}
\qquad
\begin{tikzcd}
& \star \arrow[dr, "b"] & \\
\star \arrow[ur, "b"] \arrow[rr, equal, bend right] \arrow[rr, phantom, "\Downarrow \beta"] & & \star
\end{tikzcd}
$$

HIGHER-DIMENSIONAL REWRITING SYSTEMS

Notice that the "alphabet" for a 2-dimensional rewriting system is a graph

$$\Sigma \quad = \quad \Sigma_0 \overset{s_0}{\underset{t_0}{\leftleftarrows}} \Sigma_1$$

i.e. a 1-dimensional rewriting system.

An inductive definition

Notice that the "alphabet" for a 2-dimensional rewriting system is a graph

$$\Sigma \quad = \quad \Sigma_0 \underset{t_0}{\overset{s_0}{\leftleftarrows}} \Sigma_1$$

i.e. a 1-dimensional rewriting system.

Otherwise said, from a graph we can either

- take the quotient set

  $$\tilde{\Sigma} \quad = \quad \Sigma_0 / \overset{*}{\leftrightarrow}$$

  and get a presentation

An inductive definition

Notice that the “alphabet” for a 2-dimensional rewriting system is a graph

$$\Sigma \quad = \quad \Sigma_0 \underset{t_0}{\overset{s_0}{\leftleftarrows}} \Sigma_1$$

i.e. a 1-dimensional rewriting system.

Otherwise said, from a graph we can either

- take the quotient set

  $$\tilde{\Sigma} \quad = \quad \Sigma_0 / \overset{*}{\leftrightarrow}$$

  and get a presentation
- or generate a free category

  $$\Sigma^* \quad = \quad \Sigma_0 \underset{t_0^*}{\overset{s_0^*}{\leftleftarrows}} \Sigma_1^*$$

  whose morphisms will be “terms” for the next level

A $(n+1)$-dimensional rewriting system presents an *n*-**category**

A $(n+1)$-dimensional rewriting system presents an *n*-**category**

0-category:

$x\cdot \qquad \cdot z \qquad \cdot u$

$\cdot y \qquad \cdot v$

1-categories

A $(n+1)$-dimensional rewriting system presents
an *n*-**category**

1-category:

$$\begin{tikzcd} x \arrow[rr, "g\circ f"] \arrow[dr, "f"'] & & z & & u \arrow[d, "h"] \\ & y \arrow[ur, "g"'] & & & v \end{tikzcd}$$

2-categories

A $(n+1)$-dimensional rewriting system presents
an n-**category**

2-category:

$$\begin{tikzcd}
x \arrow[rr, "i"] \arrow[dr, "f"'] & & z & u \arrow[d, "h"] \\
& y \arrow[ur, "g"'] \arrow[u, phantom, "\alpha\Downarrow"] & & v
\end{tikzcd}$$

2-categories

A $(n+1)$-dimensional rewriting system presents
an *n*-**category**

There are two compositions in a 2-category:

- vertical:

$$
\begin{tikzcd}
x \arrow[r, bend left=60, "f"] \arrow[r, "g" description] \arrow[r, bend right=60, "h"'] \arrow[r, phantom, shift left=3ex, "\alpha\Downarrow"] \arrow[r, phantom, shift right=3ex, "\beta\Downarrow"] & y
\end{tikzcd}
\quad \rightsquigarrow \quad
\begin{tikzcd}
x \arrow[r, bend left, "f"] \arrow[r, bend right, "h"'] \arrow[r, phantom, "\beta \circ \alpha \Downarrow"] & y
\end{tikzcd}
$$

- horizontal:

$$
\begin{tikzcd}
x \arrow[r, bend left, "f"] \arrow[r, bend right, "g"'] \arrow[r, phantom, "\alpha \Downarrow"] & y \arrow[r, bend left, "h"] \arrow[r, bend right, "i"'] \arrow[r, phantom, "\beta \Downarrow"] & z
\end{tikzcd}
\quad \rightsquigarrow \quad
\begin{tikzcd}
x \arrow[r, bend left, "f \otimes h"] \arrow[r, bend right, "g \otimes i"'] \arrow[r, phantom, "\alpha \otimes \beta \Downarrow"] & z
\end{tikzcd}
$$

A $(n+1)$-dimensional rewriting system presents
an *n*-**category**

The two compositions are compatible:

$$
\begin{tikzcd}
x \arrow[r, bend left, "f"] \arrow[r, "g"'] \arrow[r, phantom, bend left=20, "\alpha\Downarrow"] & y \arrow[r, "h"] \arrow[r, bend right, "i"'] \arrow[r, phantom, bend right=20, "\beta\Downarrow"] & z
\end{tikzcd}
\quad = \quad
\begin{tikzcd}
x \arrow[r, bend left, "f"] \arrow[r, bend right, "g"'] \arrow[r, phantom, "\alpha\Downarrow"] & y \arrow[r, bend left, "h"] \arrow[r, bend right, "i"'] \arrow[r, phantom, "\beta\Downarrow"] & z
\end{tikzcd}
\quad = \quad
\begin{tikzcd}
x \arrow[r, "f"] \arrow[r, bend right, "g"'] \arrow[r, phantom, bend right=20, "\alpha\Downarrow"] & y \arrow[r, bend left, "h"] \arrow[r, "i"'] \arrow[r, phantom, bend left=20, "\beta\Downarrow"] & z
\end{tikzcd}
$$

i.e.

$$(\mathrm{id}_g \otimes \beta) \circ (\alpha \otimes \mathrm{id}_h) \quad = \quad (\alpha \otimes \mathrm{id}_i) \circ (\mathrm{id}_f \otimes \beta)$$

A $(n+1)$-dimensional rewriting system presents
an ***n*-category**

n-category:

- 0-cells
- 1-cells
- …
- *n*-cells

together with

- *n* ways of composing them

A 0-signature

Σ_0

Example

signature

x *y*

Higher-dimensional rewriting systems

A 1-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

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

Higher-dimensional rewriting systems

A 1-signature = a 1-rewriting system

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

Example

signature

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

Higher-dimensional rewriting systems

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

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

Higher-dimensional rewriting systems

A 2-rewriting system

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

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

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

rules

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

Higher-dimensional rewriting systems

A 2-signature = a 2-rewriting system

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

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

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

Higher-dimensional rewriting systems

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

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

A 3-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

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

Higher-dimensional rewriting systems

Again, the general idea is that from an n-dimensional rewriting system

$$\Sigma \quad = \quad (\Sigma_0, \Sigma_1, \ldots, \Sigma_n)$$

we can do the following.

- Σ **generates** a free n-category

$$\Sigma^*$$

Higher-dimensional rewriting systems

Again, the general idea is that from an n-dimensional rewriting system

$$\Sigma \quad = \quad (\Sigma_0, \Sigma_1, \ldots, \Sigma_n)$$

we can do the following.

- Σ **generates** a free n-category

$$\Sigma^*$$

- Σ **presents** an $(n-1)$ category

$$\overline{\Sigma^*}$$

  by identifying two $(n-1)$-cells in Σ^* which are related by an n-cell

Higher-dimensional rewriting systems

Again, the general idea is that from an n-dimensional rewriting system

$$\Sigma \quad = \quad (\Sigma_0, \Sigma_1, \ldots, \Sigma_n)$$

we can do the following.

- Σ **generates** a free n-category

$$\Sigma^*$$

- Σ **presents** an $(n-1)$ category

$$\overline{\Sigma^*}$$

  by identifying two $(n-1)$-cells in Σ^* which are related by an n-cell
- we can define an $(n+1)$**-dimensional rewriting system** by specifying a set of rules

$$\Sigma_{n+1}$$

  together with their source and targets $s_n, t_n : \Sigma_{n+1} \to \Sigma_n^*$

Previous example, again

For instance, consider again the presentation of $\mathbb{N} \times (\mathbb{N}/2\mathbb{N})$:

- $\Sigma_0 = \{\star\}$
- $\Sigma_1 = \{a : \star \to \star \, , \, b : \star \to \star\}$
- $\Sigma_2 = \left\{ \begin{tikzcd} & \star \arrow[dr, "a"] & \\ \star \arrow[ur, "b"] \arrow[dr, "a"'] & \Downarrow \alpha & \star \\ & \star \arrow[ur, "b"'] & \end{tikzcd} , \begin{tikzcd} & \star \arrow[dr, "b"] & \\ \star \arrow[ur, "b"] \arrow[rr, bend right, equal] & \Downarrow \beta & \star \end{tikzcd} \right\}$

It presents the category corresponding to $\mathbb{N} \times (\mathbb{N}/2\mathbb{N})$

For instance, consider again the presentation of $\mathbb{N} \times (\mathbb{N}/2\mathbb{N})$:

- $\Sigma_0 = \{\star\}$
- $\Sigma_1 = \{a : \star \to \star\ ,\ b : \star \to \star\}$
- $$\Sigma_2 = \left\{ \begin{tikzcd} & \star \arrow[dr, "a"] \arrow[dd, phantom, "\Downarrow\alpha"] & \\ \star \arrow[ur, "b"] \arrow[dr, "a"'] & & \star \\ & \star \arrow[ur, "b"'] & \end{tikzcd} ,\ \begin{tikzcd} & \star \arrow[dr, "b"] \arrow[d, phantom, "\Downarrow\beta"] & \\ \star \arrow[ur, "b"] \arrow[rr, equal, bend right] & {} & \star \end{tikzcd} \right\}$$

It presents the category corresponding to $\mathbb{N} \times (\mathbb{N}/2\mathbb{N})$, and generates a 2-category whose 2-cells are formal (vertical and horizontal) composites of the above generators.

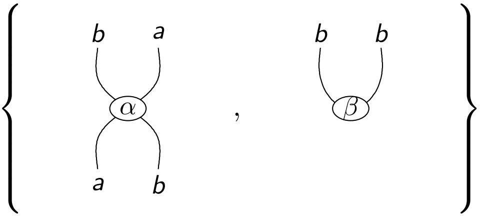
Previous example, again

For instance, consider again the presentation of $\mathbb{N} \times (\mathbb{N}/2\mathbb{N})$:

- $\Sigma_0 = \{\star\}$
- $\Sigma_1 = \{a : \star \to \star\ ,\ b : \star \to \star\}$
- $\Sigma_2 = \left\{ \begin{tikzcd} & \star \arrow[dr, "a"] & \\ \star \arrow[ur, "b"] \arrow[dr, "a"'] & \Downarrow \alpha & \star \\ & \star \arrow[ur, "b"'] & \end{tikzcd} ,\ \begin{tikzcd} & \star \arrow[dr, "b"] & \\ \star \arrow[ur, "b"] \arrow[rr, bend right, equal] & \Downarrow \beta & \star \end{tikzcd} \right\}$

It presents the category corresponding to $\mathbb{N} \times (\mathbb{N}/2\mathbb{N})$, and generates a 2-category whose 2-cells are formal (vertical and horizontal) composites of the above generators.

$\Sigma_2 =$



PRESENTING THE SIMPLICIAL CATEGORY

Let’s have a look at some examples of
3-dimensional rewriting systems
presenting *2-categories*

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

3-dimensional rewriting systems

Let's have a look at some examples of
3-dimensional rewriting systems
presenting *2-categories*

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

We will see that

- they generalize term rewriting systems
  and (some) graph rewriting systems

3-dimensional rewriting systems

Let's have a look at some examples of
3-dimensional rewriting systems
presenting *2-categories*

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

We will see that

- they generalize term rewriting systems
  and (some) graph rewriting systems
- the extra generality already brings in new problems:
  a finite rs can have an infinite number of critical pairs

The 2-category Δ

For instance, consider the 2-category Δ with

- 0-cells: $\{\star\}$
- 1-cells: $\mathbb{N}$
- a 2-cell

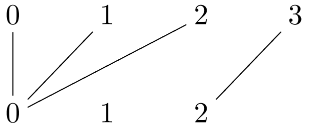
$$\begin{tikzcd} \star \arrow[r, bend left, "m", ""{name=U, below}] \arrow[r, bend right, "n"', ""{name=D, above}] & \star \arrow[from=U, to=D, phantom, "\Downarrow f"] \end{tikzcd}$$

  is an increasing function

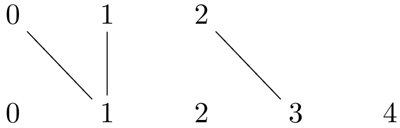
$$f \quad : \quad \{0, \ldots, m-1\} \quad \to \quad \{0, \ldots, n-1\}$$

Vertical composition is the usual composition:

$$\begin{tikzcd} \star \arrow[r, bend left, "4"] \arrow[r, bend right, "3"'] \arrow[r, phantom, "\Downarrow f"] & \star \end{tikzcd} \quad = $$

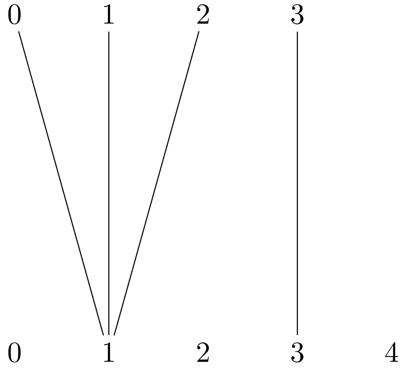


$$\begin{tikzcd} \star \arrow[r, bend left, "3"] \arrow[r, bend right, "5"'] \arrow[r, phantom, "\Downarrow g"] & \star \end{tikzcd} \quad = $$



Vertical composition is the usual composition:

$$\begin{tikzcd} \star \arrow[r, bend left, "4"] \arrow[r, bend right, "3"'] \arrow[r, phantom, "\Downarrow g \circ f"] & \star \end{tikzcd} \quad =$$



The 2-category Δ

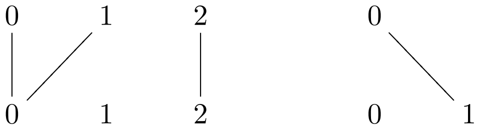
Composition of 1-cells is given by addition:

$$\star \xrightarrow{4} \star \xrightarrow{3} \star \quad \rightsquigarrow \quad \star \xrightarrow{\quad 7 \quad} \star$$

The 2-category Δ

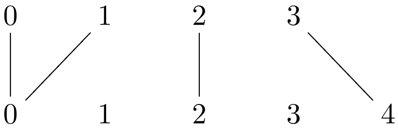
Horizontal composition is given by putting side by side:

$$\begin{tikzcd} \star \arrow[r, bend left=50, "3"{name=U}] \arrow[r, bend right=50, "3"'{name=D}] \arrow[from=U, to=D, phantom, "\Downarrow f"] & \star \end{tikzcd} \qquad \begin{tikzcd} \star \arrow[r, bend left=50, "1"{name=U}] \arrow[r, bend right=50, "2"'{name=D}] \arrow[from=U, to=D, phantom, "\Downarrow g"] & \star \end{tikzcd}$$



Horizontal composition is given by putting side by side:

$$
\begin{tikzcd}
\star \arrow[r, bend left, "4"] \arrow[r, bend right, "5"'] \arrow[r, phantom, "\Downarrow f \otimes g"] & \star
\end{tikzcd}
$$



Presenting the 2-category Δ

Σ_0

We define

- $\Sigma_0 = \{\star\}$

Presenting the 2-category Δ

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

We define

- $\Sigma_0 = \{\star\}$
- $\Sigma_1 = \left\{ \star \xrightarrow{1} \star \right\}$

Presenting the 2-category Δ

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

We define

- $\Sigma_0 = \{\star\}$
- $\Sigma_1 = \left\{ \star \xrightarrow{1} \star \right\}$, so that $\Sigma_1^* \cong \mathbb{N}$

Presenting the 2-category Δ

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

We define

- $\Sigma_0 = \{\star\}$
- $\Sigma_1 = \left\{ \star \xrightarrow{1} \star \right\}$, so that $\Sigma_1^* \cong \mathbb{N}$
- $\Sigma_2 = \left\{ \begin{tikzcd} & \star \arrow[dr, "1"] & \\ \star \arrow[ur, "1"] \arrow[rr, bend right, "1"'] \arrow[rr, phantom, "\Downarrow\mu"] & & \star \end{tikzcd}, \begin{tikzcd} \star \arrow[r, bend left=60, equal] \arrow[r, bend right=60, "1"'] \arrow[r, phantom, "\Downarrow\eta"] & \star \end{tikzcd} \right\}$

Presenting the 2-category Δ

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

We define

- $\Sigma_0 = \{\star\}$
- $\Sigma_1 = \left\{ \star \xrightarrow{1} \star \right\}$, so that $\Sigma_1^* \cong \mathbb{N}$
- $\Sigma_2 = \left\{ \begin{tikzcd} & \star \arrow[dr, "1"] & \\ \star \arrow[ur, "1"] \arrow[rr, bend right, "1"'] \arrow[rr, phantom, "\Downarrow \mu"] & & \star \end{tikzcd}, \begin{tikzcd} \star \arrow[rr, equal, bend left] \arrow[rr, bend right, "1"'] \arrow[rr, phantom, "\Downarrow \eta"] & & \star \end{tikzcd} \right\}$

Presenting the 2-category Δ

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

We define

- $\Sigma_0 = \{\star\}$
- $\Sigma_1 = \left\{ \star \xrightarrow{1} \star \right\}$, so that $\Sigma_1^* \cong \mathbb{N}$
- $\Sigma_2 = \left\{ \begin{tikzcd} & \star \arrow[dr, "1"] & \\ \star \arrow[ur, "1"] \arrow[rr, bend right, "1"'] \arrow[rr, phantom, "\Downarrow \mu"] & & \star \end{tikzcd}, \begin{tikzcd} \star \arrow[r, bend left, equal] \arrow[r, bend right, "1"'] \arrow[r, phantom, "\Downarrow \eta"] & \star \end{tikzcd} \right\}$
- $\Sigma_3 = \left\{ \begin{tikzcd} & \star \arrow[r, "1"] \arrow[d, phantom, "\Downarrow\mu"] & \star \arrow[dr, "1"] \arrow[d, phantom, "\Downarrow\mu"] & \\ \star \arrow[ur, "1"] \arrow[urr] \arrow[rrr, bend right, "1"'] & & & \star \end{tikzcd} \overset{A}{\Rrightarrow} \begin{tikzcd} & \star \arrow[r, "1"] \arrow[drr] \arrow[d, phantom, "\Downarrow\mu"] & \star \arrow[dr, "1"] \arrow[d, phantom, "\Downarrow\mu"] & \\ \star \arrow[ur, "1"] \arrow[rrr, bend right, "1"'] & & & \star \end{tikzcd}, \ldots \right\}$

Presenting the 2-category Δ

- Σ_2: the 2-generators are

$$
\begin{tikzcd}
 & \star \arrow[dr, bend left, "1"] & \\
\star \arrow[ur, bend left, "1"] \arrow[rr, bend right, "1"'] \arrow[rr, phantom, "\Downarrow \mu"] & & \star
\end{tikzcd}
\qquad\qquad
\begin{tikzcd}
\star \arrow[r, equal, bend left=50] \arrow[r, bend right=50, "1"'] \arrow[r, phantom, "\Downarrow \eta"] & \star
\end{tikzcd}
$$

- Σ_3: the rules are

$$
\begin{tikzcd}
 & \star \arrow[r, "1"] \arrow[d, phantom, "\Downarrow\mu"] & \star \arrow[dr, bend left, "1"] \arrow[d, phantom, "\Downarrow\mu"] & \\
\star \arrow[ur, bend left, "1"] \arrow[urr, bend right] \arrow[rrr, bend right, "1"'] & & & \star
\end{tikzcd}
\overset{A}{\Rrightarrow}
\begin{tikzcd}
 & \star \arrow[r, "1"] \arrow[drr, bend right] \arrow[d, phantom, "\Downarrow\mu"] & \star \arrow[dr, bend left, "1"] \arrow[d, phantom, "\Downarrow\mu"] & \\
\star \arrow[ur, bend left, "1"] \arrow[rrr, bend right, "1"'] & & & \star
\end{tikzcd}
$$

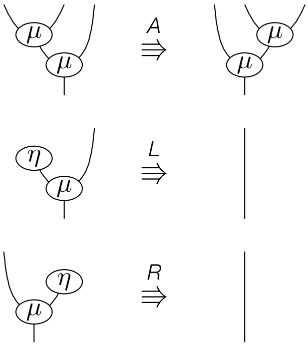
$$
\begin{tikzcd}
 & \star \arrow[dr, bend left, "1"] \arrow[d, phantom, "\Downarrow\mu"] & \\
\star \arrow[ur, equal, bend left, "1"] \arrow[ur, bend right] \arrow[u, phantom, "\Downarrow\eta"] \arrow[rr, bend right, "1"'] & & \star
\end{tikzcd}
\overset{L}{\Rrightarrow}
\begin{tikzcd}
\star \arrow[r, bend left=40, "1"] \arrow[r, bend right=40, "1"'] \arrow[r, phantom, "\parallel"] & \star
\end{tikzcd}
$$

$$
\begin{tikzcd}
 & \star \arrow[dr, equal, bend left, "1"] \arrow[dr, bend right] \arrow[d, phantom, "\Downarrow\mu"] & \\
\star \arrow[ur, bend left, "1"] \arrow[rr, bend right, "1"'] & \arrow[ur, phantom, "\Downarrow\eta"] & \star
\end{tikzcd}
\overset{R}{\Rrightarrow}
\begin{tikzcd}
\star \arrow[r, bend left=40, "1"] \arrow[r, bend right=40, "1"'] \arrow[r, phantom, "\parallel"] & \star
\end{tikzcd}
$$

Presenting the 2-category Δ

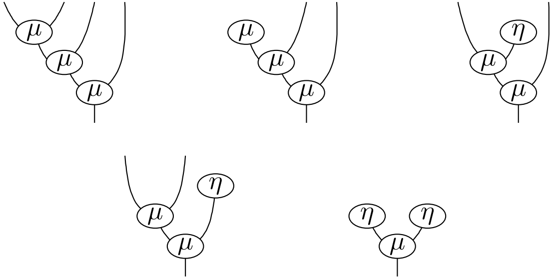
- Σ_2: the 2-generators are

- Σ_3: the rules are



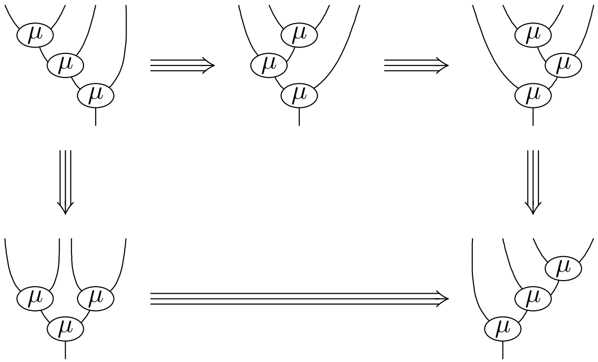
Presenting the 2-category Δ

The system is easily shown to be terminating
and the five critical pairs are confluent:



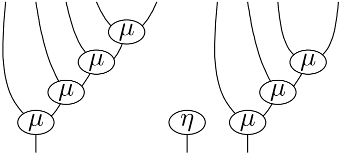
Presenting the 2-category Δ

The system is easily shown to be terminating and the five critical pairs are confluent:

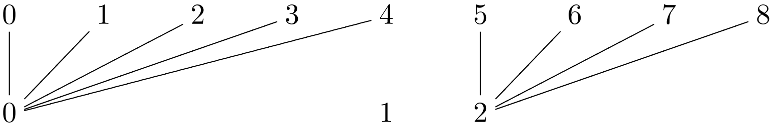


Presenting the 2-category Δ

Normal forms are horizontal composites of right combs:

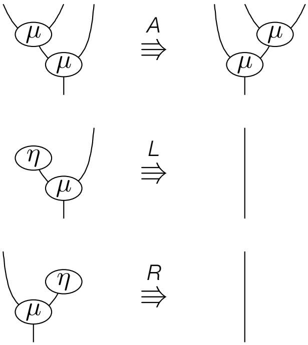


and are obviously in bijection with increasing functions:



The theory for monoids

Δ is thus the theory for monoids!



(we come back to this in a second)

Towards term rewriting systems

Notice that the previous presentation encodes the term rewriting system on the signature with

- m of arity 2
- e of arity 0

and three rules

$$m(m(x,y),z) \to m(x,m(y,z)) \qquad m(e,x) \to x \qquad m(x,e) \to x$$

Towards term rewriting systems

Notice that the previous presentation encodes the term rewriting system on the signature with

- m of arity 2
- e of arity 0

and three rules

$$m(m(x,y),z) \to m(x,m(y,z)) \qquad m(e,x) \to x \qquad m(x,e) \to x$$

It is easy to see that rewriting systems operating on terms which are linear can be directly seen as a 2-dimensional rewriting system.

What about the general case?

Towards term rewriting systems

We can also notice that our framework is more general than term rewriting systems since it allows for operations with "coarities" different from one:

The theory for monoids

Consider the category **Set**:

- it is a cartesian category:
  it has products $A \times B$ and a terminal object 1,

The theory for monoids

Consider the category **Set**:

- it is a cartesian category:
  it has products $A \times B$ and a terminal object 1,
- it can thus be seen as a 2-category:
  - 0-cells: $\star$
  - 1-cells: sets
  - 2-cells: functions

The theory for monoids

Consider the category **Set**:

- it is a cartesian category:
  it has products $A \times B$ and a terminal object 1,
- it can thus be seen as a 2-category:
  - 0-cells: $\star$
  - 1-cells: sets
  - 2-cells: functions
  - vertical composition: usual composition

$$
\begin{tikzcd}[column sep=huge]
\star \arrow[r, bend left=60, "A"{name=U}] \arrow[r, "B" description, ""{name=M}] \arrow[r, bend right=60, "C"'{name=D}] & \star
\arrow[from=U, to=M, phantom, "\Downarrow f"]
\arrow[from=M, to=D, phantom, "\Downarrow g"]
\end{tikzcd}
\quad \rightsquigarrow \quad
\begin{tikzcd}[column sep=huge]
\star \arrow[r, bend left=50, "A"{name=U}] \arrow[r, bend right=50, "C"'{name=D}] & \star
\arrow[from=U, to=D, phantom, "\Downarrow g \circ f"]
\end{tikzcd}
$$

The theory for monoids

Consider the category **Set**:

- it is a cartesian category:
  it has products $A \times B$ and a terminal object 1,
- it can thus be seen as a 2-category:
  - 0-cells: $\star$
  - 1-cells: sets
  - 2-cells: functions
  - horizontal composition: cartesian product

$$
\begin{tikzcd}
\star \arrow[r, bend left, "A"] \arrow[r, bend right, "A'"'] \arrow[r, phantom, "\Downarrow f"] & \star \arrow[r, bend left, "B"] \arrow[r, bend right, "B'"'] \arrow[r, phantom, "\Downarrow g"] & \star
\end{tikzcd}
\quad \rightsquigarrow \quad
\begin{tikzcd}
\star \arrow[rr, bend left, "A \times B"] \arrow[rr, bend right, "A' \times B'"'] \arrow[rr, phantom, "\Downarrow f \times g"] & & \star
\end{tikzcd}
$$

The theory for monoids

Consider the category **Set**:

- it is a cartesian category:
  it has products $A \times B$ and a terminal object 1,
- it can thus be seen as a 2-category:
  - 0-cells: $\star$
  - 1-cells: sets
  - 2-cells: functions

A 2-functor $F : \Delta \to$ **Set** is characterized by

- $F(\star) = \star$

The theory for monoids

Consider the category **Set**:

- it is a cartesian category:
  it has products $A \times B$ and a terminal object 1,
- it can thus be seen as a 2-category:
  - 0-cells: $\star$
  - 1-cells: sets
  - 2-cells: functions

A 2-functor $F : \Delta \to$ **Set** is characterized by

- $F(\star) = \star$
- $F(1) = A$

Consider the category **Set**:

- it is a cartesian category:
  it has products $A \times B$ and a terminal object 1,
- it can thus be seen as a 2-category:
  - 0-cells: $\star$
  - 1-cells: sets
  - 2-cells: functions

A 2-functor $F : \Delta \to$ **Set** is characterized by

- $F(\star) = \star$
- $F(1) = A$
- $F(\mu) : A \times A \to A$ and $F(\eta) : 1 \to A$

The theory for monoids

Consider the category **Set**:

- it is a cartesian category:
  it has products $A \times B$ and a terminal object 1,
- it can thus be seen as a 2-category:
  - 0-cells: $\star$
  - 1-cells: sets
  - 2-cells: functions

A 2-functor $F : \Delta \to$ **Set** is characterized by

- $F(\star) = \star$
- $F(1) = A$
- $F(\mu) : A \times A \to A$ and $F(\eta) : 1 \to A$
- such that

$$\mu \circ (\mu \times \mathrm{id}_A) = \mu \circ (\mathrm{id}_A \times \mu) \qquad \mu \circ (\eta \times \mathrm{id}_A) = \mathrm{id}_A = \mu \circ (\mathrm{id}_A \times \eta)$$

The theory for monoids

Consider the category **Set**:

- it is a cartesian category:
  it has products $A \times B$ and a terminal object 1,
- it can thus be seen as a 2-category:
  - 0-cells: $\star$
  - 1-cells: sets
  - 2-cells: functions

A 2-functor $F : \Delta \to$ **Set** is characterized by

- $F(\star) = \star$
- $F(1) = A$
- $F(\mu) : A \times A \to A$ and $F(\eta) : 1 \to A$
- such that

$$\mu \circ (\mu \times \mathrm{id}_A) = \mu \circ (\mathrm{id}_A \times \mu) \qquad \mu \circ (\eta \times \mathrm{id}_A) = \mathrm{id}_A = \mu \circ (\mathrm{id}_A \times \eta)$$

- defining F is thus the same as defining a monoid in **Set**

More generally, given a 2-category $\mathcal{C}$, the following are the same:

- a 2-functor $\Delta \to \mathcal{C}$
- a **monoid** in $\mathcal{C}$

The theory for monoids

More generally, given a 2-category $\mathcal{C}$, the following are the same:

- a 2-functor $\Delta \to \mathcal{C}$
- a **monoid** in $\mathcal{C}$

For instance, the category **Cat** of categories and functors is cartesian and can thus be considered as a 2-category:

2-functor $\Delta \to$ **Cat**

=

monoid in **Cat**

=

strict monoidal category

A **monoidal category** $(\mathcal{C}, \otimes, I, \alpha, \lambda, \rho)$ consists of

- a category $\mathcal{C}$
- a functor

$$\otimes \quad : \quad \mathcal{C} \times \mathcal{C} \quad \to \quad \mathcal{C}$$

- an object $I \in \mathcal{C}$
- invertible natural transformations

$$\alpha_{A,B,C} : (A \otimes B) \otimes C \to A \otimes (B \otimes C)$$
$$\lambda_A : I \otimes A \to A \qquad \rho_A : I \otimes A \to A$$

Monoidal categories

A **monoidal category** $(\mathcal{C}, \otimes, I, \alpha, \lambda, \rho)$ consists of

- a category $\mathcal{C}$
- a functor

$$\otimes \quad : \quad \mathcal{C} \times \mathcal{C} \quad \to \quad \mathcal{C}$$

- an object $I \in \mathcal{C}$
- invertible natural transformations

$$\alpha_{A,B,C} : (A \otimes B) \otimes C \to A \otimes (B \otimes C)$$
$$\lambda_A : I \otimes A \to A \qquad\qquad \rho_A : I \otimes A \to A$$

- such that two diagrams commute:

$$\begin{tikzcd}
((A \otimes B) \otimes C) \otimes D \arrow[r] \arrow[d] & (A \otimes (B \otimes C)) \otimes D \arrow[r] & A \otimes ((B \otimes C) \otimes D) \arrow[d] \\
(A \otimes B) \otimes (C \otimes D) \arrow[rr] & & A \otimes (B \otimes (C \otimes D))
\end{tikzcd}$$

A **monoidal category** $(\mathcal{C}, \otimes, I, \alpha, \lambda, \rho)$ consists of

- a category $\mathcal{C}$
- a functor

$$\otimes \quad : \quad \mathcal{C} \times \mathcal{C} \quad \to \quad \mathcal{C}$$

- an object $I \in \mathcal{C}$
- invertible natural transformations

$$\alpha_{A,B,C} : (A \otimes B) \otimes C \to A \otimes (B \otimes C)$$
$$\lambda_A : I \otimes A \to A \qquad\qquad \rho_A : I \otimes A \to A$$

- such that two diagrams commute:

$$\begin{tikzcd}
(A \otimes I) \otimes B \arrow[rr, "\alpha_{A,I,B}"] \arrow[dr, "\rho_A \otimes B"'] & & A \otimes (I \otimes B) \arrow[dl, "A \otimes \lambda_B"] \\
& A \otimes B &
\end{tikzcd}$$

Monoidal categories

A **monoidal category** $(\mathcal{C}, \otimes, I, \alpha, \lambda, \rho)$ consists of

- a category $\mathcal{C}$
- a functor

$$\otimes \quad : \quad \mathcal{C} \times \mathcal{C} \quad \to \quad \mathcal{C}$$

- an object $I \in \mathcal{C}$
- invertible natural transformations

$$\alpha_{A,B,C} : (A \otimes B) \otimes C \to A \otimes (B \otimes C)$$

$$\lambda_A : I \otimes A \to A \qquad\qquad \rho_A : I \otimes A \to A$$

- such that two diagrams commute:

$$\begin{tikzcd}
(A \otimes I) \otimes B \arrow[rr, "\alpha_{A,I,B}"] \arrow[dr, "\rho_A \otimes B"'] & & A \otimes (I \otimes B) \arrow[dl, "A \otimes \lambda_B"] \\
& A \otimes B &
\end{tikzcd}$$

It is **strict** when α, λ and ρ are identity natural transformations.

If we consider the 3-category $\mathbf{Cat}_2$ with

- 0-cells: $\{\star\}$
- 1-cells: categories
- 2-cells: functors
- 3-cells: invertible natural transformations

The theory of 2-monoids

If we consider the 3-category $\mathbf{Cat}_2$ with

- 0-cells: $\{\star\}$
- 1-cells: categories
- 2-cells: functors
- 3-cells: invertible natural transformations

it is easy to adapt previous work in order to find a 4-rewriting system Σ such that

$$\text{functors } \overline{\Sigma}^* \to \mathbf{Cat}_2 \quad \cong \quad \text{monoidal categories}$$

The theory of 2-monoids

If we consider the 3-category **Cat**$_2$ with

- 0-cells: $\{\star\}$
- 1-cells: categories
- 2-cells: functors
- 3-cells: invertible natural transformations

it is easy to adapt previous work in order to find a 4-rewriting system Σ such that

$$\text{functors } \overline{\Sigma}^* \to \mathbf{Cat}_2 \quad \cong \quad \text{monoidal categories}$$

Namely,

- $\Sigma_0 = \{\star\}$
- $\Sigma_1 = \{1\}$
- $\Sigma_2 = \{\mu, \eta\}$
- $\Sigma_3 = \{A, L, R\}$
- $\Sigma_4 = \{$the two axioms of monoidal categories$\}$

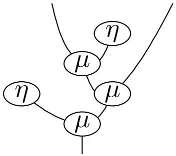
$$\text{functors } \overline{\Sigma}^* \to \mathbf{Cat}_2 \quad \cong \quad \text{monoidal categories}$$

Namely,

- $\Sigma_0 = \{\star\}$
- $\Sigma_1 = \{1\}$
- $\Sigma_2 = \{\mu, \eta\}$
- $\Sigma_3 = \{A, L, R\}$
- $\Sigma_4 = \{$the two axioms of monoidal categories$\}$

This means that a formula can be seen as a 2-cell in Σ^*:

$$I \otimes ((A \otimes I) \otimes B) \quad \rightsquigarrow$$



and a natural transformation as a 3-cell in Σ^*.

Can we use rewriting theory
to show something interesting?

Mac Lane's coherence theorem

Theorem (Mac Lane)

Every diagram built from the morphisms

$$\alpha_{A,B,C} \qquad \lambda_A \qquad \rho_A \qquad \alpha^{-1}_{A,B,C} \qquad \lambda^{-1}_A \qquad \rho^{-1}_A$$

by composing and tensoring commutes in a monoidal category.

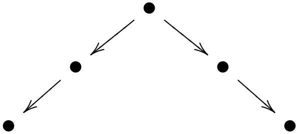
$$\begin{tikzcd}
(A \otimes I) \otimes (B \otimes I) \arrow[r] \arrow[d] & \ldots \arrow[d] \\
\ldots \arrow[r] & I \otimes (A \otimes B)
\end{tikzcd}$$

Classical tools in rewriting theory

Suppose given a terminating (string / term / …) rewriting system in which critical pairs can be joined

Lemma (Newman)

The rewriting system is confluent:



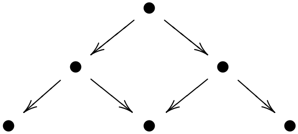
where tiles are either independent rewritings or critical pairs in context.

Classical tools in rewriting theory

Suppose given a terminating (string / term / …) rewriting system in which critical pairs can be joined

Lemma (Newman)

The rewriting system is confluent:



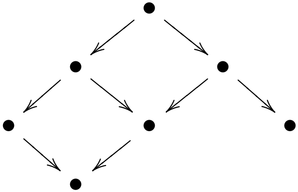
where tiles are either independent rewritings or critical pairs in context.

Classical tools in rewriting theory

Suppose given a terminating (string / term / ...) rewriting system in which critical pairs can be joined

Lemma (Newman)

The rewriting system is confluent:



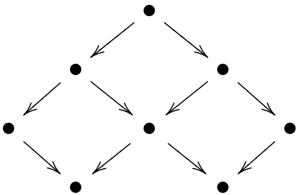
where tiles are either independent rewritings or critical pairs in context.

Classical tools in rewriting theory

Suppose given a terminating (string / term / …) rewriting system in which critical pairs can be joined

Lemma (Newman)

The rewriting system is confluent:



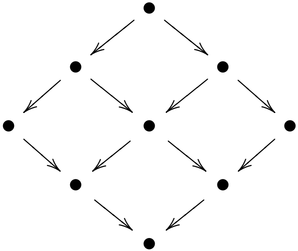
where tiles are either independent rewritings or critical pairs in context.

Classical tools in rewriting theory

Suppose given a terminating (string / term / …) rewriting system in which critical pairs can be joined

Lemma (Newman)

The rewriting system is confluent:



where tiles are either independent rewritings or critical pairs in context.

Classical tools in rewriting theory

Suppose given a terminating (string / term / …) rewriting system in which critical pairs can be joined

Lemma (Church-Rosser)

Two terms which are convertible ($t \overset{*}{\leftrightarrow} u$) can be joined:

$$\begin{tikzcd}
& \bullet \arrow[dl] \arrow[dr] & & \bullet \arrow[dl] \arrow[dr] & \\
t & & \bullet & & u
\end{tikzcd}$$

where tiles are either independent rewritings or critical pairs in context.

Classical tools in rewriting theory

Suppose given a terminating (string / term / …) rewriting system in which critical pairs can be joined

Lemma (Church-Rosser)

Two terms which are convertible ($t \overset{*}{\leftrightarrow} u$) can be joined:

$$
\begin{tikzcd}
 & \bullet \arrow[dl] \arrow[dr] & & \bullet \arrow[dl] \arrow[dr] & \\
t \arrow[dr] & & \bullet \arrow[dl] & & u \\
 & \bullet & & &
\end{tikzcd}
$$

where tiles are either independent rewritings or critical pairs in context.

Suppose given a terminating (string / term / …) rewriting system in which critical pairs can be joined

Lemma (Church-Rosser)

Two terms which are convertible ($t \overset{*}{\leftrightarrow} u$) can be joined:

$$
\begin{tikzcd}
 & \bullet \arrow[dl] \arrow[dr] & & \bullet \arrow[dl] \arrow[dr] & \\
t \arrow[dr] & & \bullet \arrow[dl] \arrow[dr] & & u \arrow[dl] \\
 & \bullet & & \bullet &
\end{tikzcd}
$$

where tiles are either independent rewritings or critical pairs in context.

Classical tools in rewriting theory

Suppose given a terminating (string / term / …) rewriting system in which critical pairs can be joined

Lemma (Church-Rosser)

Two terms which are convertible ($t \overset{*}{\leftrightarrow} u$) can be joined:

$$
\begin{tikzcd}
& \bullet \arrow[dl] \arrow[dr] & & \bullet \arrow[dl] \arrow[dr] & \\
t \arrow[dr] & & \bullet \arrow[dl] \arrow[dr] & & u \arrow[dl] \\
& \bullet \arrow[dr] & & \bullet \arrow[dl] & \\
& & \bullet & &
\end{tikzcd}
$$

where tiles are either independent rewritings or critical pairs in context.

Suppose given a terminating (string / term / …) rewriting system in which critical pairs can be joined

Lemma

In particular, if we have $\hat{t} \overset{*}{\leftrightarrow} u \overset{*}{\leftrightarrow} \hat{t}$ where $\hat{t}$ is in normal form,

$$
\begin{tikzcd}
u \arrow[d, bend right=60, leftrightarrow, "*"'] \arrow[d, bend left=60, leftrightarrow, "*"] \\
\hat{t}
\end{tikzcd}
$$

it can be paved with tiles corresponding to either independent rewritings or critical pairs in context.

A variant of the theory for 2-monoids

Now, consider the following 4-rewriting system

- $\Sigma_0 = \{\star\}$
- $\Sigma_1 = \{1\}$
- $\Sigma_2 = \{\mu, \eta\}$
- $\Sigma_3 = \{A, L, R\}$
- $\Sigma_4 = \{$one rule for each of the five critical pairs for monoids$\}$

functor $\overline{\Sigma}^* \to$ **Cat**

=

monoidal category satisfying three more axioms

we have shown that $(\Sigma_0, \ldots, \Sigma_3)$ *is terminating and confluent*

A variant of the theory for 2-monoids

Now, consider the following 4-rewriting system

- $\Sigma_0 = \{\star\}$
- $\Sigma_1 = \{1\}$
- $\Sigma_2 = \{\mu, \eta\}$
- $\Sigma_3 = \{A, L, R\}$
- $\Sigma_4 = \{$one rule for each of the five critical pairs for monoids$\}$

A diagram

$$\begin{tikzcd}
(A \otimes I) \otimes (B \otimes I) \arrow[r] \arrow[d] & \dots \arrow[d] \\
\dots \arrow[r] & I \otimes (A \otimes B)
\end{tikzcd}$$

corresponds to a pair of 3-cells in Σ^* in which cells can be used in both directions.

A variant of the theory for 2-monoids

Now, consider the following 4-rewriting system

- $\Sigma_0 = \{\star\}$
- $\Sigma_1 = \{1\}$
- $\Sigma_2 = \{\mu, \eta\}$
- $\Sigma_3 = \{A, L, R\}$
- $\Sigma_4 = \{$one rule for each of the five critical pairs for monoids$\}$

Consider a pair of 3-cells in Σ^*

$$
\begin{tikzcd}
u & g \arrow[l, Rightarrow] \arrow[d, Rightarrow] \\
f \arrow[u, Rightarrow] \arrow[r, Rightarrow] & t
\end{tikzcd}
$$

A variant of the theory for 2-monoids

Now, consider the following 4-rewriting system

- $\Sigma_0 = \{\star\}$
- $\Sigma_1 = \{1\}$
- $\Sigma_2 = \{\mu, \eta\}$
- $\Sigma_3 = \{A, L, R\}$
- $\Sigma_4 = \{$one rule for each of the five critical pairs for monoids$\}$

Consider a pair of 3-cells in Σ^*

$$\begin{tikzcd} u \arrow[r, Leftrightarrow, "g"] \arrow[d, Leftrightarrow, "f"'] & t \\ t & \end{tikzcd}$$

A variant of the theory for 2-monoids

Now, consider the following 4-rewriting system

- $\Sigma_0 = \{\star\}$
- $\Sigma_1 = \{1\}$
- $\Sigma_2 = \{\mu, \eta\}$
- $\Sigma_3 = \{A, L, R\}$
- $\Sigma_4 = \{$one rule for each of the five critical pairs for monoids$\}$

Consider a pair of 3-cells in Σ^*

$$
\begin{tikzcd}
u \arrow[r, Leftrightarrow, "g"] \arrow[d, Leftrightarrow, "f"'] & t \arrow[d, Rightarrow, "h"] \\
t \arrow[r, Rightarrow, "h"'] & \hat{t}
\end{tikzcd}
$$

where $h : t \to \hat{t}$ is a 3-cell going to the normal form of t

A variant of the theory for 2-monoids

Now, consider the following 4-rewriting system

- $\Sigma_0 = \{\star\}$
- $\Sigma_1 = \{1\}$
- $\Sigma_2 = \{\mu, \eta\}$
- $\Sigma_3 = \{A, L, R\}$
- $\Sigma_4 = \{$one rule for each of the five critical pairs for monoids$\}$

Consider a pair of 3-cells in Σ^*

$$\begin{tikzcd}
u \arrow[r, Leftrightarrow, "g"] \arrow[d, Leftrightarrow, "f"'] & t \arrow[d, Rightarrow, "h"] \\
t \arrow[r, Rightarrow, "h"'] & \hat{t}
\end{tikzcd}$$

by Church-Rosser lemma it can be filled by 4-cells

A variant of the theory for 2-monoids

Now, consider the following 4-rewriting system

- $\Sigma_0 = \{\star\}$
- $\Sigma_1 = \{1\}$
- $\Sigma_2 = \{\mu, \eta\}$
- $\Sigma_3 = \{A, L, R\}$
- $\Sigma_4 = \{$one rule for each of the five critical pairs for monoids$\}$

Consider a pair of 3-cells in Σ^*

$$\begin{tikzcd}
u \arrow[r, Leftrightarrow, "g"] \arrow[d, Leftrightarrow, "f"'] & t \arrow[d, Rightarrow, "h"] & \\
t \arrow[r, Rightarrow, "h"'] & \hat{t} \arrow[r, Rightarrow, "h^{-1}"'] & t
\end{tikzcd}$$

by Church-Rosser lemma it can be filled by 4-cells

A variant of the theory for 2-monoids

Now, consider the following 4-rewriting system

- $\Sigma_0 = \{\star\}$
- $\Sigma_1 = \{1\}$
- $\Sigma_2 = \{\mu, \eta\}$
- $\Sigma_3 = \{A, L, R\}$
- $\Sigma_4 = \{$one rule for each of the five critical pairs for monoids$\}$

Consider a pair of 3-cells in Σ^*

$$
\begin{tikzcd}
u & & g \arrow[ll, Rightarrow] \arrow[d, Rightarrow] \\
f \arrow[u, Rightarrow] \arrow[rr, Rightarrow] & & t
\end{tikzcd}
$$

It can be filled by 4-cells!

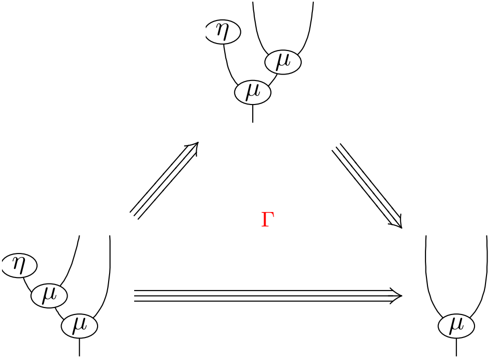
Removing superfluous axioms

We have almost shown Mac Lane coherence theorem excepting that we have five 4-cells instead of two.

Removing superfluous axioms

We have almost shown Mac Lane coherence theorem excepting that we have five 4-cells instead of two.

For instance, why is the following 4-cell superfluous?



Removing superfluous axioms

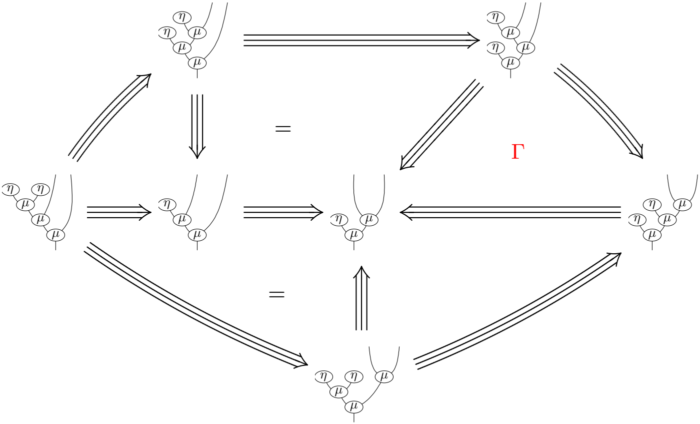
We have almost shown Mac Lane coherence theorem excepting that we have five 4-cells instead of two.

Consider the following critical triple:

Removing superfluous axioms

We have almost shown Mac Lane coherence theorem excepting that we have five 4-cells instead of two.

Consider the following critical triple:



Removing superfluous axioms

We have almost shown Mac Lane coherence theorem excepting that we have five 4-cells instead of two.

Consider the following critical triple:

and Γ is therefore superfluous!

Tietze transformations

Theorem (Tietze)

Consider two string rewriting systems (Σ, R) and (Σ', R'). They present the same monoid, i.e.

$$\Sigma^* / \overset{*}{\leftrightarrow}_R \qquad \equiv \qquad \Sigma'^* / \overset{*}{\leftrightarrow}_{R'}$$

if and only if we can obtain one from the other by the following transformations and their inverses

1. *adding a superfluous generator*

$$(\Sigma, R) \qquad \rightsquigarrow \qquad (\Sigma \uplus a, R \uplus \{a \to u\})$$

with $u \in \Sigma^*$

2. *adding a superfluous relation*

$$(\Sigma, R) \qquad \rightsquigarrow \qquad (\Sigma, R \uplus \{u \to v\})$$

such that $u \overset{*}{\leftrightarrow}_R v$.

Theorem (Tietze)

Two presentations of the same monoid differ by

1. *adding/removing superfluous generators*
2. *adding/removing superfluous relations*

We can thus

Tietze transformations

Theorem (Tietze)

Two presentations of the same monoid differ by

1. *adding/removing superfluous generators*
2. *adding/removing superfluous relations*

We can thus

- investigate Tietze transformations for higher-dimensional rewriting systems
  (Gaussent, Guiraud, Malbos)

Tietze transformations

Theorem (Tietze)

Two presentations of the same monoid differ by

1. *adding/removing superfluous generators*
2. *adding/removing superfluous relations*

We can thus

- investigate Tietze transformations for higher-dimensional rewriting systems
  (Gaussent, Guiraud, Malbos)
- refine Knuth-Bendix completion algorithm
  - keep track of coherence cells during the completion
  - use the fact that we can add not only superfluous relations but also generators

  (Guiraud, Malbos, Mimram)

PRESENTING THE CATEGORY OF BIJECTIONS

(or not)

Presenting the 2-category **Bij**

Another very interesting example was studied by Lafont.

Presenting the 2-category **Bij**

Another very interesting example was studied by Lafont.

Consider the 2-category **Bij** defined similarly as Δ:

- 0-cells: $\{\star\}$
- 1-cells: $\mathbb{N}$
- 2-cells: *bijective* functions

$$f \quad : \quad \{0,\ldots,m-1\} \quad \to \quad \{0,\ldots,n-1\}$$

Presenting the 2-category **Bij**

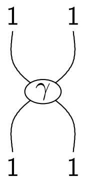
Since we know that bijections can be expressed as products of transpositions, it can be expected that the 2-category **Bij** admits the following presentation:

- $\Sigma_0 = \{\star\}$

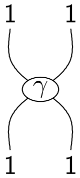
Since we know that bijections can be expressed as products of transpositions, it can be expected that the 2-category **Bij** admits the following presentation:

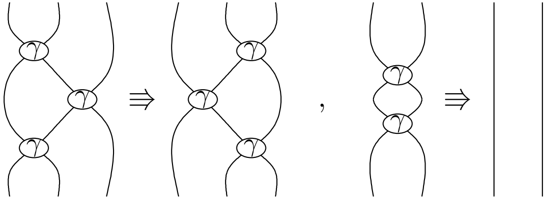
- $\Sigma_0 = \{\star\}$
- $\Sigma_1 = \{1 : \star \to \star\}$

Since we know that bijections can be expressed as products of transpositions, it can be expected that the 2-category **Bij** admits the following presentation:

- $\Sigma_0 = \{\star\}$
- $\Sigma_1 = \{1 : \star \to \star\}$
- $\Sigma_2 = \Big\{$  $\Big\}$

Since we know that bijections can be expressed as products of transpositions, it can be expected that the 2-category **Bij** admits the following presentation:

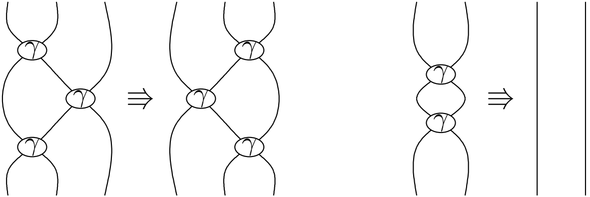
- $\Sigma_0 = \{\star\}$
- $\Sigma_1 = \{1 : \star \to \star\}$
- $\Sigma_2 = \Bigg\{$ 

$\Bigg\}$
- $\Sigma_3 = \Bigg\{$ 

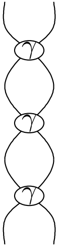
$\Bigg\}$

Presenting the 2-category **Bij**

The rules

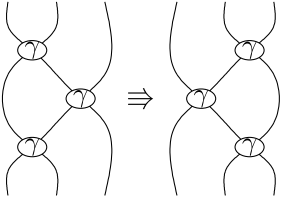


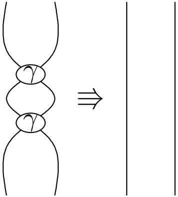
induce confluent critical pairs:



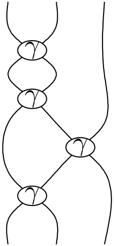
Presenting the 2-category **Bij**

The rules



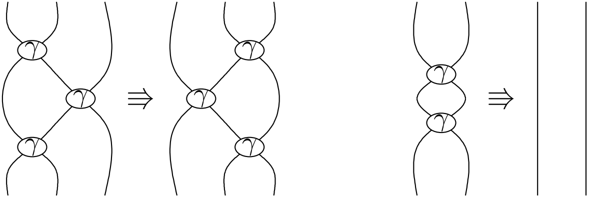


induce confluent critical pairs:

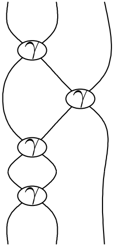


Presenting the 2-category **Bij**

The rules

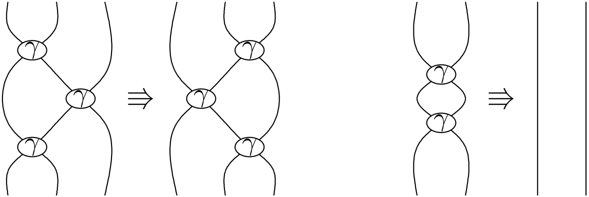


induce confluent critical pairs:

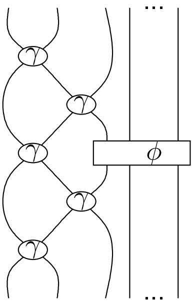


Presenting the 2-category **Bij**

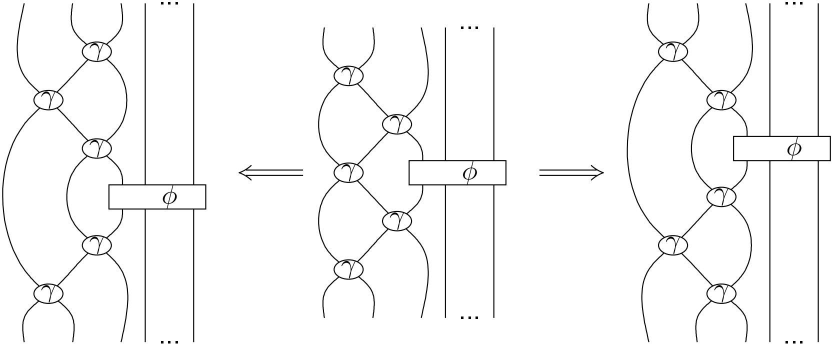
The rules



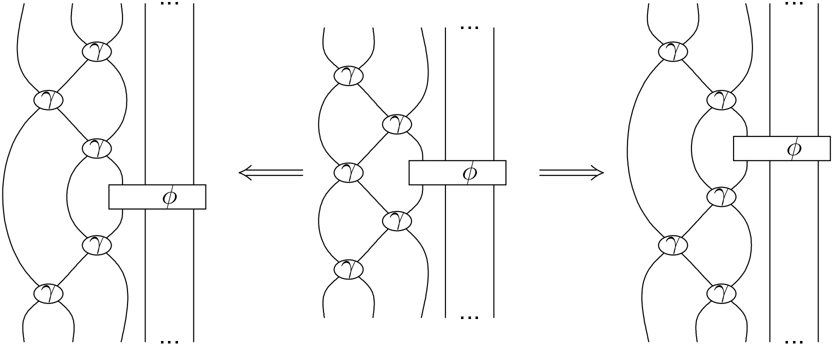
induce an infinite number of critical pairs:



A family of critical pairs



A family of critical pairs



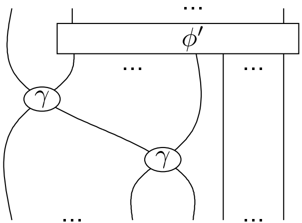
And we cannot reduce further with a generic ϕ!

Showing confluence

Confluence can however be shown as follows.

Confluence can however be shown as follows.

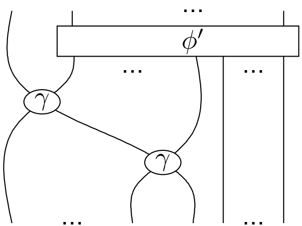
1. A 2-cell ϕ is in *canonical form* when it is either an identity or of the form



Showing confluence

Confluence can however be shown as follows.

1. A 2-cell ϕ is in *canonical form* when it is either an identity or of the form

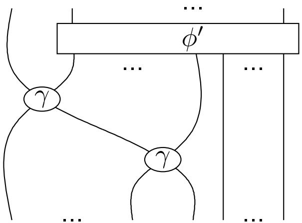


2. By induction (on the number of generators), every morphism ϕ rewrites to a morphism in canonical form.

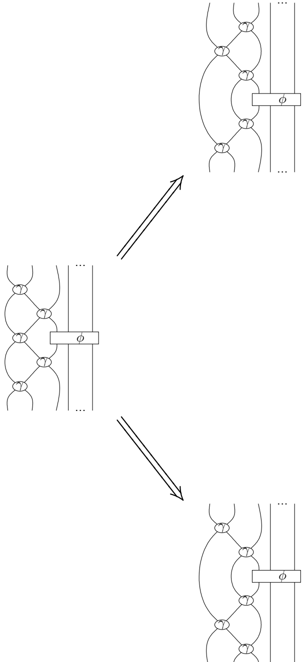
Showing confluence

Confluence can however be shown as follows.

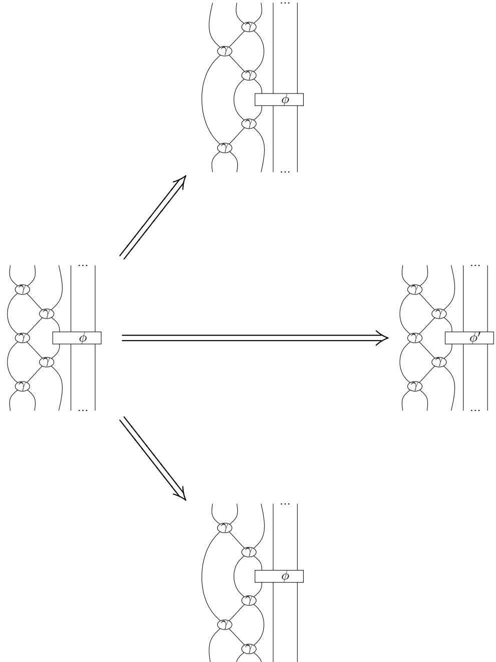
1. A 2-cell ϕ is in *canonical form* when it is either an identity or of the form

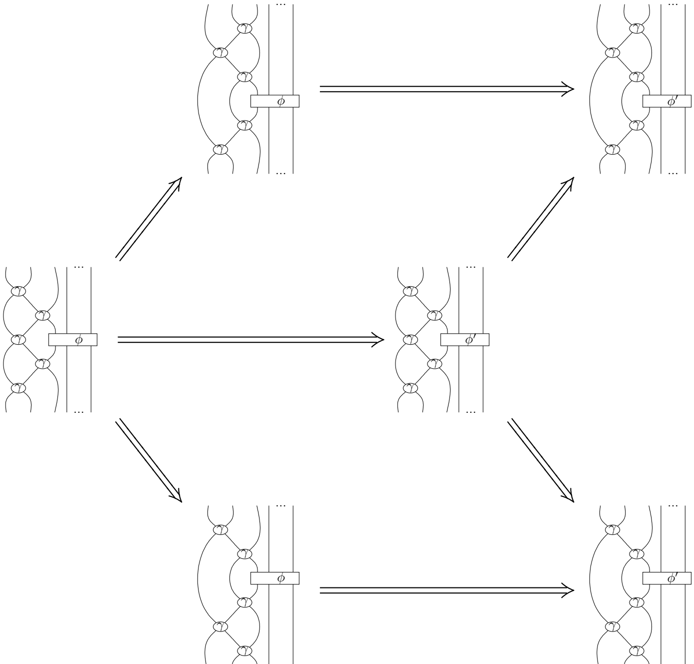


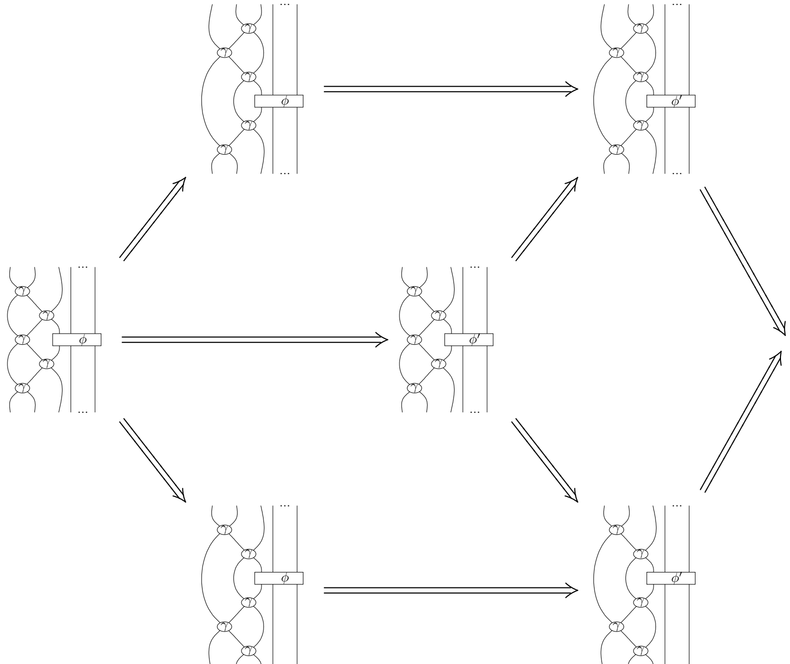
2. By induction (on the number of generators), every morphism ϕ rewrites to a morphism in canonical form.
3. For morphisms in canonical form the families of critical pairs are confluent.



γ
ϕ

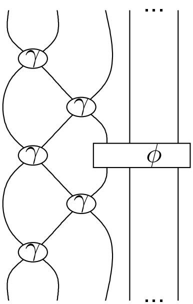

γ
ϕ
ϕ′


γ
ϕ
ϕ'



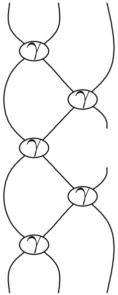
Computing critical pairs

Even though there can be an infinite number of critical pairs, I constructed a unification algorithm which is able to compute them all by generalizing the notion of diagram we consider.

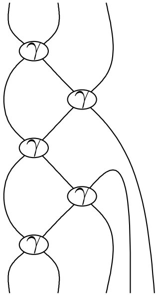


Computing critical pairs

Even though there can be an infinite number of critical pairs, I constructed a unification algorithm which is able to compute them all by generalizing the notion of diagram we consider.



Even though there can be an infinite number of critical pairs, I constructed a unification algorithm which is able to compute them all by generalizing the notion of diagram we consider.



LAWVERE THEORIES

(or not)

Suppose given a signature for terms

$$S = \{m : 2, e : 0\}$$

We can form a category S^*

- objects: $\mathbb{N}$
- morphisms $m \to n$ are
  n-uples of terms with variables in $x_1, \ldots, x_m$

$$\langle \quad m(m(x_1, x_1), x_2) \quad , \quad e \quad , \quad x_2 \quad \rangle \qquad : \qquad 2 \quad \to \quad 3$$

Categories from terms

Suppose given a signature for terms

$$S = \{m : 2, e : 0\}$$

We can form a category S^*

- objects: $\mathbb{N}$
- morphisms $m \to n$ are
  n-uples of terms with variables in $x_1, \ldots, x_m$

$$\langle \quad m(m(x_1, x_1), x_2) \quad , \quad e \quad , \quad x_2 \quad \rangle \qquad : \qquad 2 \quad \to \quad 3$$

- composition is given by substitution:

$$5 \xrightarrow{\langle t_1, t_2 \rangle} 2 \xrightarrow{\langle u_1, u_2, u_3 \rangle} 3$$

Suppose given a signature for terms

$$S = \{m : 2, e : 0\}$$

We can form a category S^*

- objects: $\mathbb{N}$
- morphisms $m \to n$ are
  n-uples of terms with variables in $x_1, \ldots, x_m$

$$\langle \quad m(m(x_1,x_1),x_2) \quad , \quad e \quad , \quad x_2 \quad \rangle \qquad : \qquad 2 \quad \to \quad 3$$

- composition is given by substitution:

$$\begin{tikzcd} 5 \arrow[r, "\langle t_1,t_2 \rangle"] \arrow[rr, bend right, "\langle u_1[\sigma],u_2[\sigma],u_3[\sigma] \rangle"'] & 2 \arrow[r, "\langle u_1,u_2,u_3 \rangle"] & 3 \end{tikzcd}$$

with $\sigma = [t_1/x_1, t_2/x_2]$

Lawvere theories

The category S^* is easily shown to be a **Lawvere theory**:

- a cartesian category,
- whose objects are integers,
- and product is given on objects by addition.

The category S^* is easily shown to be a **Lawvere theory**:

- a cartesian category,
- whose objects are integers,
- and product is given on objects by addition.

A term rewriting system

$$(S, R)$$

thus presents the Lawvere theory

$$S^* / \overset{*}{\leftrightarrow}_R$$

The category S^* is easily shown to be a **Lawvere theory**:

- a cartesian category,
- whose objects are integers,
- and product is given on objects by addition.

A term rewriting system

$$(S, R)$$

thus presents the Lawvere theory

$$S^* / \overset{*}{\leftrightarrow}_R$$

A Lawvere theory can be seen as
a 2-category with only one 0-cell

Encoding term rewriting systems

Theorem (Burroni)

Given a term rewriting system (S, R), the 3-rewriting system Σ defined by

- $\Sigma_0 = \{\star\}$
- $\Sigma_1 = \{1\}$
- $\Sigma_2 = R \uplus \{\delta : 2 \to 1, \varepsilon : 0 \to 1, \gamma : 2 \to 2\}$
- $\Sigma_3 = S \uplus \{(\delta, \varepsilon, \gamma)$ is a natural commutative comonoid$\}$

presents the same Lawvere theory.

A term rewriting system

=

a linear term rewriting system

+

explicit duplication, erasure and swapping of variables

An example: commutative monoids

Consider the term rewriting system for commutative monoids

$$
\begin{array}{rcl}
m(m(x,y),z) & \to & m(x,m(y,z)) \\
m(e,x) & \to & x \\
m(x,e) & \to & x \\
m(x,y) & \to & m(y,x)
\end{array}
$$

An example: commutative monoids

Consider the term rewriting system for commutative monoids

$$\begin{array}{rcl} m(m(x,y),z) & \to & m(x,m(y,z)) \\ m(e,x) & \to & x \\ m(x,e) & \to & x \\ m(x,y) & \to & m(y,x) \end{array}$$

It is not terminating!

An example: commutative monoids

Consider the 3-rewriting system for commutative monoids

$$
\begin{array}{rcl}
m \circ (m \otimes \mathrm{id}_1) & \to & m \circ (\mathrm{id}_1 \otimes m) \\
m \circ (\eta \otimes \mathrm{id}_1) & \to & \mathrm{id}_1 \\
m \circ (\mathrm{id}_1 \otimes \eta) & \to & \mathrm{id}_1 \\
{\color{red} m \circ \gamma} & {\color{red}\to} & {\color{red} m} \\
& \vdots &
\end{array}
$$

We get much more rules and critical pairs
but the rewriting system is terminating
and can be completed

An example: commutative monoids

From this it can be shown that the term rewriting system for commutative monoids presents the Lawvere theory whose

- objects are integers
- morphisms

  $$M \quad : \quad m \quad \to \quad n$$

  are $(m \times n)$-matrices with coefficients in $\mathbb{N}$

$$\langle\, m(m(x_1,x_1),x_2)\ ,\ e\ ,\ x_2 \,\rangle \quad : \quad 2 \to 3 \qquad\qquad \rightsquigarrow \qquad\qquad \begin{pmatrix} 2 & 0 & 0 \\ 1 & 0 & 1 \end{pmatrix}$$

An example: commutative monoids

From this it can be shown that the term rewriting system for commutative monoids presents the Lawvere theory whose

- objects are integers
- morphisms

  $$M \quad : \quad m \quad \to \quad n$$

  are $(m \times n)$-matrices with coefficients in $\mathbb{N}$

$$\begin{array}{c} \langle\, m(m(x_1,x_1),x_2)\ ,\ e\ ,\ x_2\,\rangle \\ \langle m(x_1,m(x_1,x_2))\ ,\ e\ ,\ m(e,x_2)\rangle \end{array} \quad : \quad 2 \to 3 \qquad \rightsquigarrow \qquad \begin{pmatrix} 2 & 0 & 0 \\ 1 & 0 & 1 \end{pmatrix}$$

The morale

Explicit handling of symmetries can be fruitful!

(e.g. a convergent presentation of
the theory of Frobenius algebras)

CONCLUSION

Conclusion

- rewriting theory generalizes to higher dimensions
- classical tools too
- it turns out to be powerful to
  - better understand algebraic structures (i.e. present them)
  - address coherence issues
  - it also has applications in algebraic topology:
    in order to have a structure up to homotopy, one has (roughly) to find a convergent presentation and explicitly describe all the critical n-uples
- it also brings finer understanding on traditional rewriting (Knuth-Bendix)