Categorical coherence from term rewriting systems

Samuel Mimram

FSCD conference / 4 July 2023

École polytechnique

The coherence theorem for monoidal categories

A **monoidal category** $(C, \otimes, e, \alpha, \lambda, \rho)$ comes equipped with

$$\alpha_{x,y,z} : (x \otimes y) \otimes z \overset{\sim}{\to} x \otimes (y \otimes z) \qquad \lambda_x : e \otimes x \overset{\sim}{\to} x \qquad \rho_x : x \otimes e \overset{\sim}{\to} x$$

satisfying axioms.

$$\begin{tikzcd}
((x \otimes y) \otimes z) \otimes w \arrow[r] \arrow[d] & (x \otimes (y \otimes z)) \otimes w \arrow[r] & x \otimes ((y \otimes z) \otimes w) \arrow[d] \\
(x \otimes y) \otimes (z \otimes w) \arrow[rr] & & x \otimes (y \otimes (z \otimes w))
\end{tikzcd}$$

$$\begin{tikzcd}
(x \otimes e) \otimes y \arrow[rr] \arrow[dr] & & x \otimes (e \otimes y) \arrow[dl] \\
& x \otimes y &
\end{tikzcd}$$

The coherence theorem for monoidal categories

A **monoidal category** $(C, \otimes, e, \alpha, \lambda, \rho)$ comes equipped with

$$\alpha_{x,y,z} : (x \otimes y) \otimes z \xrightarrow{\sim} x \otimes (y \otimes z) \qquad \lambda_x : e \otimes x \xrightarrow{\sim} x \qquad \rho_x : x \otimes e \xrightarrow{\sim} x$$

satisfying axioms.

The **coherence theorem** for monoidal categories states that every diagram whose morphisms are composites of α, λ and ρ commutes:

$$\begin{tikzcd}
(e \otimes x) \otimes y \arrow[r] \arrow[dd] & e \otimes (x \otimes y) \arrow[dr] & \\
& & e \otimes (x \otimes (y \otimes e)) \arrow[d] \\
x \otimes y \arrow[rr] & & x \otimes (y \otimes e)
\end{tikzcd}$$

The coherence theorems for monoidal categories

In fact, there are various ways of formulating the coherence theorem:

1. **Coherence**:
   every diagram in a free monoidal category made up of α, λ and ρ commutes.
2. **Coherence**:
   every diagram in a monoidal category made up of α, λ and ρ commutes.
3. **Strictification**:
   every monoidal category is monoidally equivalent to a strict monoidal category.
4. **Global strictification**:
   the forgetful 2-functor from strict monoidal categories to monoidal categories has a left adjoint and the components of the unit are equivalences.

The coherence theorems for symmetric monoidal categories

A monoidal category is **symmetric** when equipped with

$$\gamma_{x,y} : x \otimes y \to y \otimes x$$

satisfying axioms.

Similar coherence theorems hold but they are more subtle:

- in 2. we have to restrict to “generic” diagrams, e.g. the following diagram does not commute:

$$\begin{tikzcd} x \otimes x \arrow[r, bend left, "\gamma_{x,x}"] \arrow[r, bend right, "\mathrm{id}_{x \otimes x}"'] & x \otimes x \end{tikzcd}$$

- in 4., for a strict symmetric monoidal category, we suppose that α, λ and ρ are strict but not γ
- (global) strictification is only shown for free categories

A generic framework for coherence

Here, we investigate general coherence theorems where

- coherence holds with respect to part of the structure (e.g. α, λ and ρ but not γ)
- structural morphisms can erase or duplicate variables:

$$\delta_{x,y,z} : x \otimes (y \oplus z) \to (x \otimes y) \oplus (x \otimes z)$$

- we use rewriting theory.

A generic framework for coherence

Here, we investigate general coherence theorems where

- coherence holds with respect to part of the structure (e.g. α, λ and ρ but not γ)
- structural morphisms can erase or duplicate variables:

$$\delta_{x,y,z} : x \otimes (y \oplus z) \to (x \otimes y) \oplus (x \otimes z)$$

- we use rewriting theory.

We begin by studying the situation in an abstract setting.

Part I

Abstract coherence

An abstract setting

Fix a category $\mathcal{C}$ which we think of as describing an **algebraic structure**.

For instance, we have a theory of symmetric monoidal categories:

- the objects of $\mathcal{C}$ are formal tensor expressions

$$e \otimes ((x \otimes e) \otimes y)$$

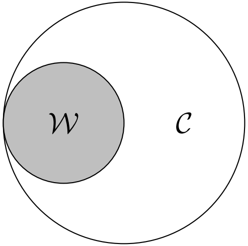
- morphisms are composites of α, λ, ρ and γ modulo axioms.

An abstract setting

Fix a category $\mathcal{C}$ which we think of as describing an **algebraic structure**.

We suppose fixed a subgroupoid $\mathcal{W} \subseteq \mathcal{C}$ with the same objects, which we are interested in strictifying.

(for SMC, $\mathcal{W}$ would be the groupoid of composites of α, λ and ρ, but not γ)



Quotient of categories

The **quotient** $\mathcal{C}/\mathcal{W}$ is the universal way of making the elements of $\mathcal{W}$ identities

$$\begin{tikzcd}
\mathcal{C} \arrow[r, "F"] \arrow[d] & \mathcal{D} \\
\mathcal{C}/\mathcal{W} \arrow[ur, dotted, "\tilde{F}"'] &
\end{tikzcd}$$

Question
When is the quotient functor $\mathcal{C} \to \mathcal{C}/\mathcal{W}$ an equivalence of categories?

Quotient of categories

The **quotient** $\mathcal{C}/\mathcal{W}$ is the universal way of making the elements of $\mathcal{W}$ identities

$$\begin{tikzcd}
\mathcal{C} \arrow[r, "F"] \arrow[d] & \mathcal{D} \\
\mathcal{C}/\mathcal{W} \arrow[ur, dotted, "\tilde{F}"'] &
\end{tikzcd}$$

Question
When is the quotient functor $\mathcal{C} \to \mathcal{C}/\mathcal{W}$ an equivalence of categories?

Intuitively, when $\mathcal{W}$ does not contain non-trivial information!

Rigid groupoids

A groupoid $\mathcal{W}$ is **rigid** when either

(i) any two parallel morphisms $f, g : x \to y$ are equal

(ii) any automorphism $f : x \to x$ is an identity

(iii) $\mathcal{W}$ is equivalent to $\bigsqcup_X \mathbf{1}$

Quotienting by rigid groupoids

When $\mathcal{W}$ is rigid the quotient $\mathcal{C}/\mathcal{W}$ has a simple description:

- objects are eq. classes of objects with $[x] = [y]$ when there is $w : x \to y$ in $\mathcal{W}$,

Quotienting by rigid groupoids

When $\mathcal{W}$ is rigid the quotient $\mathcal{C}/\mathcal{W}$ has a simple description:

- objects are eq. classes of objects with $[x] = [y]$ when there is $w : x \to y$ in $\mathcal{W}$,
- morphisms are eq. classes of morphisms with $[f] = [g]$ when there is v and w in $\mathcal{W}$ such that

$$\begin{tikzcd} x \arrow[r, "f"] \arrow[d, "\mathcal{W}\ni v"'] & y \arrow[d, "w\in\mathcal{W}"] \\ x' \arrow[r, "g"'] & y' \end{tikzcd}$$

Quotienting by rigid groupoids

When $\mathcal{W}$ is rigid the quotient $\mathcal{C}/\mathcal{W}$ has a simple description:

- objects are eq. classes of objects with $[x] = [y]$ when there is $w : x \to y$ in $\mathcal{W}$,
- morphisms are eq. classes of morphisms with $[f] = [g]$ when there is v and w in $\mathcal{W}$ such that

$$\begin{tikzcd}
x \arrow[r, "f"] \arrow[d, "\mathcal{W}\ni v"'] & y \arrow[d, "w\in\mathcal{W}"] \\
x' \arrow[r, "g"'] & y'
\end{tikzcd}$$

- we compose $[f] : [x] \to [y]$ and $[g] : [y] \to [z]$ as

$$\begin{tikzcd}
x \arrow[r, "f"] & y & \\
 & y \arrow[r, "g"] & z
\end{tikzcd}$$

Quotienting by rigid groupoids

When $\mathcal{W}$ is rigid the quotient $\mathcal{C}/\mathcal{W}$ has a simple description:

- objects are eq. classes of objects with $[x] = [y]$ when there is $w : x \to y$ in $\mathcal{W}$,
- morphisms are eq. classes of morphisms with $[f] = [g]$ when there is v and w in $\mathcal{W}$ such that

$$\begin{tikzcd}
x \arrow[r, "f"] \arrow[d, "\mathcal{W}\ni v"'] & y \arrow[d, "w\in\mathcal{W}"] \\
x' \arrow[r, "g"'] & y'
\end{tikzcd}$$

- we compose $[f] : [x] \to [y]$ and $[g] : [y] \to [z]$ as

$$\begin{tikzcd}
x \arrow[r, "f"] & y \arrow[d, "\in\mathcal{W}"] & \\
 & y \arrow[r, "g"] & z
\end{tikzcd}$$

Quotienting by rigid groupoids

~~When $\mathcal{W}$ is rigid~~ the quotient $\mathcal{C}/\mathcal{W}$ has a simple description:

- objects are eq. classes of objects with $[x] = [y]$ when there is $w : x \to y$ in $\mathcal{W}$,
- morphisms are eq. classes of morphisms with $[f] = [g]$ when there is v and w in $\mathcal{W}$ such that

$$\begin{tikzcd}
x \arrow[r, "f"] \arrow[d, "\mathcal{W}\ni v"'] & y \arrow[d, "w\in\mathcal{W}"] \\
x' \arrow[r, "g"'] & y'
\end{tikzcd}$$

- we compose $[f] : [x] \to [y]$ and $[g] : [y] \to [z]$ as

$$\begin{tikzcd}
x \arrow[r, "f"] & y \arrow[d, shift right, "\mathcal{W}\ni"'] \arrow[d, shift left, "\in\mathcal{W}"] & \\
& y \arrow[r, "g"] & z
\end{tikzcd}$$

Rigidification

The **rigidification** $\mathcal{C}/\!\!/\mathcal{W}$ of $\mathcal{W}$ in $\mathcal{C}$ is obtained from $\mathcal{C}$ by identifying any two parallel morphisms in $\mathcal{W}$.

Proposition
The quotient can be obtained is two steps: $\mathcal{C}/\mathcal{W} = (\mathcal{C}/\!\!/\mathcal{W})/\mathcal{W}$

Coherence for quotients

Theorem
The quotient functor $\mathcal{C} \to \mathcal{C}/\mathcal{W}$ is an equivalence of categories if and only if $\mathcal{W}$ is rigid.

$\mathcal{C}$	$\mathcal{C}/\mathcal{W}$
$\begin{tikzcd} x \arrow[r, shift left, "f"] \arrow[r, shift right, "g"', red] & y \end{tikzcd}$	$\begin{tikzcd} x \arrow[loop right, "f"] \end{tikzcd}$
$\begin{tikzcd} x \arrow[r, shift left, "f", red] \arrow[r, shift right, "g"', red] & y \end{tikzcd}$	x

Coherence for quotients

Theorem
The quotient functor $\mathcal{C} \to \mathcal{C}/\mathcal{W}$ is an equivalence of categories if and only if $\mathcal{W}$ is rigid.

Proof.
The quotient functor $\mathcal{C} \to \mathcal{C}/\!\!/\mathcal{W} \to \mathcal{C}$ is surjective on objects and full.

We need to show that it is faithful iff $\mathcal{W}$ is rigid.

- If the quotient functor is faithful, given $w, w' : x \to y$, we have $[w] = [w']$ and thus $w = w'$.
- If $\mathcal{W}$ is rigid, given $f, g : x \to y$ such that $[f] = [g]$, we have

$$\begin{tikzcd} x \arrow[r, "f"'] \arrow[d, "\mathcal{W}\ni v"'] & y \arrow[d, "w\in\mathcal{W}"] \\ x \arrow[r, "g"] & y \end{tikzcd}$$

By rigidity, $v = \mathrm{id}_x$ and $w = \mathrm{id}_y$. □

Coherence for algebras

An **algebra** for $\mathcal{C}$ in $\mathcal{D}$ is a functor $\mathcal{C} \to \mathcal{D}$, we write $\mathsf{Alg}(\mathcal{C}, \mathcal{D})$ for the category of algebras. In particular, we are interested in $\mathsf{Alg}(\mathcal{C}) = \mathsf{Alg}(\mathcal{C}, \mathbf{Cat})$.

Coherence for algebras

An **algebra** for $\mathcal{C}$ in $\mathcal{D}$ is a functor $\mathcal{C} \to \mathcal{D}$, we write $\mathsf{Alg}(\mathcal{C}, \mathcal{D})$ for the category of algebras. In particular, we are interested in $\mathsf{Alg}(\mathcal{C}) = \mathsf{Alg}(\mathcal{C}, \mathbf{Cat})$.

Theorem
A functor $\mathsf{F} : \mathcal{C} \to \mathcal{C}'$ *is an equivalence iff* $\mathsf{Alg}(\mathsf{F}, \mathcal{D}) : \mathsf{Alg}(\mathcal{C}, \mathcal{D}) \to \mathsf{Alg}(\mathcal{C}', \mathcal{D})$ *is an equivalence natural in* $\mathcal{D}$*.*

Proof.
Given a $\mathbf{2}$-category $\mathcal{K}$, the Yoneda functor

$$\begin{aligned} \mathsf{Y}_{\mathcal{K}} : \mathcal{K}^{\mathrm{op}} &\to [\mathcal{K}, \mathbf{Cat}] \\ \mathsf{C} &\mapsto \mathcal{K}(\mathsf{C}, -) \end{aligned}$$

is a local isomorphism. In particular, with $\mathcal{K} = \mathbf{Cat}$, we have $\mathsf{Y}_{\mathbf{Cat}}\mathcal{C} = \mathsf{Alg}(\mathcal{C}, -)$. □

Coherence for algebras

An **algebra** for $\mathcal{C}$ in $\mathcal{D}$ is a functor $\mathcal{C} \to \mathcal{D}$, we write $\mathsf{Alg}(\mathcal{C}, \mathcal{D})$ for the category of algebras. In particular, we are interested in $\mathsf{Alg}(\mathcal{C}) = \mathsf{Alg}(\mathcal{C}, \mathbf{Cat})$.

Theorem
A functor $F : \mathcal{C} \to \mathcal{C}'$ *is an equivalence iff* $\mathsf{Alg}(F, \mathcal{D}) : \mathsf{Alg}(\mathcal{C}, \mathcal{D}) \to \mathsf{Alg}(\mathcal{C}', \mathcal{D})$ *is an equivalence natural in* $\mathcal{D}$*.*

Proof.
Given a 2-category $\mathcal{K}$, the Yoneda functor

$$\begin{aligned} Y_{\mathcal{K}} : \mathcal{K}^{\mathrm{op}} &\to [\mathcal{K}, \mathbf{Cat}] \\ C &\mapsto \mathcal{K}(C, -) \end{aligned}$$

is a local isomorphism. In particular, with $\mathcal{K} = \mathbf{Cat}$, we have $Y_{\mathbf{Cat}}\mathcal{C} = \mathsf{Alg}(\mathcal{C}, -)$. ☐

Corollary
The canonical functor $\mathsf{Alg}(\mathcal{C}/\mathcal{W}) \to \mathsf{Alg}(\mathcal{C})$ *is an equivalence iff* $\mathcal{W}$ *is rigid.*

Question
How do we show rigidity in practice?

In the following, we are interested in the case where $\mathcal{C}$ is a groupoid.

Abstract rewriting systems

An **abstract rewriting system P** is a graph

$$\begin{tikzcd} & P_1 \arrow[dl, shift right, "s_0"'] \arrow[dl, shift left, "t_0"] \\ P_0 & \end{tikzcd}$$

$$\begin{tikzcd} P & = & x \arrow[r, bend left, "f"] & y \arrow[l, bend left, "g"] & z \arrow[l, "h"'] \end{tikzcd}$$

Abstract rewriting systems

An **abstract rewriting system P** is a graph

$$\begin{tikzcd}
 & P_1 \arrow[dl, shift right, "s_0"'] \arrow[dl, shift left, "t_0"] \arrow[d] \\
P_0 & P_1^{\sim} \arrow[l, shift right] \arrow[l, shift left]
\end{tikzcd}$$

It generates a groupoid with $P_1^{\sim}$ as set of morphisms.

$$\begin{tikzcd}
P & = & x \arrow[r, bend left, "f"] & y \arrow[l, bend left, "g"] & z \arrow[l, "h"']
\end{tikzcd}$$

$$x \xrightarrow{f} y \xrightarrow{g} x \xrightarrow{f} y \xrightarrow{h^-} z$$

Abstract rewriting systems

An **extended abstract rewriting system** P is a graph

$$\begin{tikzcd}
 & P_1 \arrow[dl, shift right, "s_0"'] \arrow[dl, shift left, "t_0"] \arrow[d] & P_2 \arrow[dl, shift right, "s_1"'] \arrow[dl, shift left, "t_1"] \\
P_0 & P_1^{\sim} \arrow[l, shift right] \arrow[l, shift left] &
\end{tikzcd}$$

together with a set of 2-cells

$$\begin{tikzcd}
 & x \arrow[dl, bend right, "f"'] \arrow[dr, bend left, "g"] & \\
y \arrow[dr, bend right, "h^{-}"'] & \overset{A}{\Rightarrow} & y \arrow[dl, bend left, "h^{-}"] \\
 & z &
\end{tikzcd}$$

Abstract rewriting systems

An **extended abstract rewriting system** P is a graph

$$\begin{tikzcd}
 & P_1 \arrow[dl, shift right, "s_0"'] \arrow[dl, shift left, "t_0"] \arrow[d] & P_2 \arrow[dl, shift right, "s_1"'] \arrow[dl, shift left, "t_1"] \\
P_0 & P_1^{\sim} \arrow[l, shift right] \arrow[l, shift left] &
\end{tikzcd}$$

together with a set of **2**-cells

$$\begin{tikzcd}
 & x \arrow[dl, bend right, "f"'] \arrow[dr, bend left, "g"] & \\
y \arrow[dr, bend right, "h^{-}"'] & \overset{A}{\Rightarrow} & y \arrow[dl, bend left, "h^{-}"] \\
 & z &
\end{tikzcd}$$

It **presents** a groupoid $\overline{P} = P^{\sim} / \sim$.

Abstract rewriting systems: Tietze equivalences

In a situation such as

$$\begin{tikzcd}
 & P_1 \arrow[dl, shift right, "s_0"'] \arrow[dl, shift left, "t_0"] \arrow[d] & P_2 \arrow[dl, shift right, "s_1"'] \arrow[dl, shift left, "t_1"] \\
P_0 & P_1^{\sim} \arrow[l, shift right] \arrow[l, shift left] &
\end{tikzcd}$$

with

$$\begin{tikzcd}
 & x \arrow[dl, bend right, "f"'] \arrow[dr, bend left, "g"] & \\
y \arrow[dr, bend right, "h^-"'] & \overset{A}{\Rightarrow} & y \arrow[dl, bend left, "h^-"] \\
 & z &
\end{tikzcd}$$

- if A can be derived from other elements P_2, we can remove it,
- if we remove $f \in P_1$ and $A \in P_2$ the presented groupoid is the same.

Abstract rewriting systems

Suppose given an extended ARS P together with $W \subseteq \mathsf{P}_1$.

We say that P is **W-convergent** when it has

- termination: there is no infinite sequence of morphisms in W

$$\begin{tikzcd} x_0 \arrow[r, "f_0"] & x_1 \arrow[r, "f_1"] & x_2 \arrow[r, "f_2"] & \cdots \end{tikzcd}$$

- local confluence:

$$\begin{tikzcd} & x \arrow[dl, "W\ni"'] \arrow[dr, "\in W"] & \\ y_1 \arrow[dr, dotted, "W\ni *"'] & \in \mathsf{P}_2^{\sim} & y_2 \arrow[dl, dotted, "* \in W"] \\ & z & \end{tikzcd}$$

Abstract rewriting systems

By adapting standard rewriting techniques,

Lemma ("Newman")
If P *is* W*-convergent then it is* W*-confluent:*

$$\begin{tikzcd}
 & x \arrow[dl, "W\ni"'] \arrow[dr, "\in W"] & \\
y_1 \arrow[dr, dotted, "W\ni *"'] & \in \mathsf{P}_2^{\sim} & y_2 \arrow[dl, dotted, "*\in W"] \\
 & z &
\end{tikzcd}$$

Abstract rewriting systems

By adapting standard rewriting techniques,

Lemma (“Newman”)
If P *is* W*-convergent then it is* W*-confluent:*

Lemma (“Church-Rosser”)
If P *is* W*-convergent then for any two parallel* W*-morphisms in* $\overline{\mathsf{P}}$ *are equal.*

Proof.

$$\begin{tikzcd}
x \arrow[r, "p_1^-"] \arrow[d, "n_x"'] & y_1 \arrow[r, "q_1^+"] \arrow[d, dashed, "n_{y_1}"] & x_2 \arrow[r] \arrow[d, dashed, "n_{x_2}"] & \cdots \arrow[r] & x_n \arrow[r, "p_n^-"] \arrow[d, dashed, "n_{x_n}"] & y_n \arrow[r, "q_n^-"] \arrow[d, dashed, "n_{y_n}"] & y \arrow[d, dashed, "n_y"] \\
\hat{x} \arrow[r, equal] & \hat{x} \arrow[r, equal] & \hat{x} \arrow[r, equal] & \cdots \arrow[r, equal] & \hat{x} \arrow[r, equal] & \hat{x} \arrow[r, equal] & \hat{x}
\end{tikzcd}$$

□

Abstract rewriting systems

Theorem
If P *is* W*-convergent then the groupoid generated by* W *in* $\overline{\mathsf{P}}$ *is rigid.*

Abstract rewriting systems

Theorem
If P *is* W*-convergent then the groupoid generated by* W *in* $\overline{\mathsf{P}}$ *is rigid.*

Writing $N(\overline{\mathsf{P}})$ for the full subcategory of $\overline{\mathsf{P}}$ whose objects are normal forms (are not the source of a morphism in W),

Theorem
If (P, W) *is* W*-convergent then* $\overline{P}/W \cong N(\overline{\mathsf{P}})$*.*

Summary

Given (P, W), we have shown that the following definitions of **coherence** of P wrt W are equivalent:

(i) Every parallel zig-zags with edges in W are equal
(i.e. the subgroupoid of $\overline{P}$ generated by W is rigid).

(ii) The quotient map $\overline{\mathsf{P}} \to \overline{\mathsf{P}}/W$ is an equivalence of categories.

(iii) The inclusion $\mathsf{Alg}(\overline{\mathsf{P}}/W) \to \mathsf{Alg}(\overline{\mathsf{P}})$ is an equivalence of categories.

(iv) The canonical morphism $N(\mathsf{P}) \to \overline{\mathsf{P}}$ is an equivalence.

Part II

Coherence from term rewriting systems

From ARS to TRS

In order to obtain result about actual categorical structures, we need to go from ARS to term rewriting systems!

Term rewriting systems

A **term rewriting system P** consists of

- P_1: operations with arities
- P_2: equations between generated terms

Example

The TRS **Mon** for **monoids** is

$$\left\langle \begin{array}{r} m : 2 \\ e : 0 \end{array} \;\middle|\; \begin{array}{l} \alpha : m(m(x,y),z) = m(x,m(y,z)) \\ \lambda : \qquad m(e,x) = x \\ \rho : \qquad m(x,e) = x \end{array} \right\rangle$$

Term rewriting systems

An **extended term rewriting system P** consists of

- P_1: operations with arities
- P_2: 2-generators between generated terms
- P_3: equations between 2-generators

Example
The 2-TRS **Mon** for **monoids** is

$$\left\langle \begin{array}{r} m : 2 \\ e : 0 \end{array} \;\middle|\; \begin{array}{rr} \alpha : & m(m(x,y),z) \Rightarrow m(x,m(y,z)) \\ \lambda : & m(e,x) \Rightarrow x \\ \rho : & m(x,e) \Rightarrow x \end{array} \;\middle|\; \begin{tikzcd} {} \arrow[r] \arrow[d] & {} \arrow[r] \arrow[d, phantom, "A"] & {} \arrow[d] \\ {} \arrow[rr] & & {} \end{tikzcd} \qquad \begin{tikzcd} {} \arrow[rr] \arrow[dr] & {} \arrow[d, phantom, "U"] & {} \arrow[dl] \\ & {} & \end{tikzcd} \right\rangle$$

Term rewriting systems

An **extended term rewriting system** P consists of

- P$_1$: operations with arities
- P$_2$: 2-generators between generated terms
- P$_3$: equations between 2-generators

Example

The 2-TRS **Mon** for **monoids** is

$$\left\langle \begin{array}{r} m : 2 \\ e : 0 \end{array} \;\middle|\; \begin{array}{ll} \alpha : & m(m(x,y),z) \Rightarrow m(x,m(y,z)) \\ \lambda : & \qquad m(e,x) \Rightarrow x \\ \rho : & \qquad m(x,e) \Rightarrow x \end{array} \;\middle|\; \begin{tikzcd} {} \arrow[r] \arrow[d] \arrow[drr, phantom, "A"] & {} \arrow[r] & {} \arrow[d] \\ {} \arrow[rr] & & {} \end{tikzcd} \qquad \begin{tikzcd} {} \arrow[rr] \arrow[dr] & {} \arrow[d, phantom, "U"] & {} \arrow[dl] \\ & {} & \end{tikzcd} \right\rangle$$

NB: fixing m and n, P induces an abstract rewriting systems on terms $m \to n$.

Lawvere theories

A **Lawvere theory** $\mathcal{T}$ is a cartesian category such that objects are integers with cartesian product given by addition.

Lawvere theories

A **2-Lawvere theory** $\mathcal{T}$ is a cartesian 2-category with invertible 2-cells such that objects are integers with cartesian product given by addition.

Lawvere theories

A **2-Lawvere theory** $\mathcal{T}$ is a cartesian 2-category with invertible 2-cells such that objects are integers with cartesian product given by addition.

Any 2-TRS P induces a 2-LT $\overline{\mathsf{P}}$ with

- morphisms $\langle t_1, \ldots, t_n \rangle : m \to n$ are n-tuples of terms with m variables
- 2-cells are generated by 2-generators, quotiented by equations

Lawvere theories

A **2-Lawvere theory** $\mathcal{T}$ is a cartesian 2-category with invertible 2-cells such that objects are integers with cartesian product given by addition.

Any 2-TRS $\mathbf{P}$ induces a 2-LT $\overline{\mathbf{P}}$ with

- morphisms $\langle t_1, \ldots, t_n \rangle : m \to n$ are n-tuples of terms with m variables
- 2-cells are generated by 2-generators, quotiented by equations

An **algebra** for $\mathcal{T}$ is a product-preserving 2-functor $\mathcal{T} \to \mathbf{Cat}$.

Example
An algebra for $\overline{\mathbf{Mon}}$ is a monoidal category.

Algebras for Mon

With **Mon** being

$$\left\langle \begin{array}{r} m : 2 \\ e : 0 \end{array} \;\middle|\; \begin{array}{ll} \alpha : & m(m(x,y),z) = m(x,m(y,z)) \\ \lambda : & \qquad\quad m(e,x) = x \\ \rho : & \qquad\quad m(x,e) = x \end{array} \;\middle|\; \begin{tikzcd} {} \arrow[r] \arrow[d] & {} \arrow[r] \arrow[d, phantom, "A"] & {} \arrow[d] \\ {} \arrow[rr] & & {} \end{tikzcd} \quad \begin{tikzcd} {} \arrow[rr] \arrow[dr] \arrow[rr, phantom, "U"'] & & {} \arrow[dl] \\ & {} & \end{tikzcd} \right\rangle$$

A functor $F : \overline{\mathsf{Mon}} \to \mathbf{Cat}$ consists of

- a category $\mathcal{C} = F1$

Algebras for Mon

With **Mon** being

$$\left\langle \begin{array}{r} m : 2 \\ e : 0 \end{array} \;\middle|\; \begin{array}{rr} \alpha : & m(m(x,y),z) \Rightarrow m(x,m(y,z)) \\ \lambda : & m(e,x) \Rightarrow x \\ \rho : & m(x,e) \Rightarrow x \end{array} \;\middle|\; \begin{tikzcd} {} \arrow[r] \arrow[d] & {} \arrow[r] \arrow[d, phantom, "A"] & {} \arrow[d] \\ {} \arrow[rr] & & {} \end{tikzcd} \quad \begin{tikzcd} {} \arrow[rr] \arrow[dr] & {} \arrow[d, phantom, "U"] & {} \arrow[dl] \\ & {} & \end{tikzcd} \right\rangle$$

A functor $F : \overline{\mathbf{Mon}} \to \mathbf{Cat}$ consists of

- a category $\mathcal{C} = F1$
- thus $Fn = \mathcal{C}^n$

Algebras for Mon

With **Mon** being

$$\left\langle \begin{array}{r} m : 2 \\ e : 0 \end{array} \;\middle|\; \begin{array}{rr} \alpha : & m(m(x,y),z) \Rightarrow m(x,m(y,z)) \\ \lambda : & m(e,x) \Rightarrow x \\ \rho : & m(x,e) \Rightarrow x \end{array} \;\middle|\; \begin{tikzcd} {} \arrow[r] \arrow[d] & {} \arrow[r] & {} \arrow[d] & & {} \arrow[rr] \arrow[dr] & U & {} \arrow[dl] \\ {} \arrow[rr] & A & {} & & & {} & \end{tikzcd} \right\rangle$$

A functor $F : \overline{\text{Mon}} \to \textbf{Cat}$ consists of

- a category $\mathcal{C} = F1$
- thus $Fn = \mathcal{C}^n$
- two functors

$$\otimes = Fm : \mathcal{C}^2 \to \mathcal{C} \qquad\qquad I = Fe : 1 \to \mathcal{C}$$

Algebras for Mon

With **Mon** being

$$\left\langle \begin{array}{r} m : 2 \\ e : 0 \end{array} \middle| \begin{array}{rl} \alpha : & m(m(x,y),z) \Rightarrow m(x,m(y,z)) \\ \lambda : & \qquad m(e,x) \Rightarrow x \\ \rho : & \qquad m(x,e) \Rightarrow x \end{array} \middle| \begin{tikzcd} {} \arrow[r] \arrow[d] & {} \arrow[r] \arrow[d, phantom, "A"] & {} \arrow[d] \\ {} \arrow[rr] & & {} \end{tikzcd} \quad \begin{tikzcd} {} \arrow[rr] \arrow[dr] & {} \arrow[d, phantom, "U"] & {} \arrow[dl] \\ & {} & \end{tikzcd} \right\rangle$$

A functor $F : \overline{\text{Mon}} \to \mathbf{Cat}$ consists of

- a category $\mathcal{C} = F1$
- thus $Fn = \mathcal{C}^n$
- two functors

$$\otimes = Fm : \mathcal{C}^2 \to \mathcal{C} \qquad\qquad I = Fe : 1 \to \mathcal{C}$$

- satisfying the axioms of monoidal categories

Quotienting by rigid subgroupoids

Fix a 2-TRS P with a subset $W \subseteq \mathsf{P}_2$ of 2-generators generating a $(2,1)$-category $\mathcal{W}$.

Quotienting by rigid subgroupoids

Fix a 2-TRS P with a subset $W \subseteq \mathsf{P}_2$ of 2-generators generating a $(2,1)$-category $\mathcal{W}$.

$\mathcal{W}$ is **2-rigid** when any parallel 2-cells are equal.

Quotienting by rigid subgroupoids

Fix a 2-TRS P with a subset $W \subseteq \mathsf{P}_2$ of 2-generators generating a $(2, 1)$-category $\mathcal{W}$.

$\mathcal{W}$ is **2-rigid** when any parallel 2-cells are equal.

Theorem
The quotient 2*-functor* $\overline{\mathsf{P}} \to \overline{\mathsf{P}}/W$ *is a local equivalence iff* $\mathcal{W}$ *is 2-rigid.*

Quotienting by rigid subgroupoids

Fix a 2-TRS P with a subset $W \subseteq \mathsf{P}_2$ of 2-generators generating a $(2,1)$-category $\mathcal{W}$.

$\mathcal{W}$ is **2-rigid** when any parallel 2-cells are equal.

Theorem
The quotient 2*-functor* $\overline{\mathsf{P}} \to \overline{\mathsf{P}}/W$ *is a local equivalence iff* $\mathcal{W}$ *is 2-rigid.*

Theorem
If P *is* W*-convergent then* $\mathcal{W}$ *is 2-rigid.*

Quotienting by rigid subgroupoids

Fix a 2-TRS P with a subset $W \subseteq \mathsf{P}_2$ of 2-generators generating a $(2,1)$-category $\mathcal{W}$.

$\mathcal{W}$ is **2-rigid** when any parallel 2-cells are equal.

Theorem
The quotient 2-functor $\overline{\mathsf{P}} \to \overline{\mathsf{P}}/W$ *is a local equivalence iff* $\mathcal{W}$ *is 2-rigid.*

Theorem
If P *is* W*-convergent then* $\mathcal{W}$ *is 2-rigid.*

Example
The theory **Mon** is W-convergent with W = all 2-cells.

$$\left\langle \begin{array}{r} m : 2 \\ e : 0 \end{array} \middle| \begin{array}{rl} \alpha : & m(m(x,y),z) \Rightarrow m(x,m(y,z)) \\ \lambda : & m(e,x) \Rightarrow x \\ \rho : & m(x,e) \Rightarrow x \end{array} \middle| \begin{tikzcd} {} \arrow[r] \arrow[d] & {} \arrow[r] \arrow[d, phantom, "A"] & {} \arrow[d] \\ {} \arrow[rr] & & {} \end{tikzcd} \quad \begin{tikzcd} {} \arrow[rr] \arrow[dr] & {} \arrow[d, phantom, "U"] & {} \arrow[dl] \\ & {} & \end{tikzcd} \right\rangle$$

Confluence of Mon

Note that there are 5 critical branchings:

$$\begin{tikzcd}
m(m(m(x_1,x_2),x_3),x_4) \arrow[r, "\alpha"] \arrow[dd, "\alpha"'] & m(m(x_1,m(x_2,x_3)),x_4) \arrow[dr, "\alpha"] & \\
& \overset{A}{\Rrightarrow} & m(x_1,m(m(x_2,x_3),x_4)) \arrow[d, "\alpha"] \\
m(m(x_1,x_2),m(x_3,x_4)) \arrow[rr, "\alpha"'] & & m(x_1,m(x_2,m(x_3,x_4)))
\end{tikzcd}$$

$$\begin{tikzcd}
m(m(e,x_1),x_2) \arrow[rr, "\alpha"] \arrow[dr, "\lambda"'] & \overset{B}{\Rrightarrow} & m(e,m(x_1,x_2)) \arrow[dl, "\lambda"] \\
& m(x_1,x_2) &
\end{tikzcd}$$

$$\begin{tikzcd}
m(m(x_1,e),x_2) \arrow[rr, "\alpha"] \arrow[dr, "\rho"'] & \overset{C}{\Rrightarrow} & m(x_1,m(e,x_2)) \arrow[dl, "\lambda"] \\
& m(x_1,x_2) &
\end{tikzcd}$$

$$\begin{tikzcd}
m(m(x_1,x_2),e) \arrow[rr, "\alpha"] \arrow[dr, "\rho"'] & \overset{D}{\Rrightarrow} & m(x_1,m(x_2,e)) \arrow[dl, "\rho"] \\
& m(x_1,x_2) &
\end{tikzcd}$$

$$\begin{tikzcd}
m(e,e) \arrow[d, bend right, "\lambda"'] \arrow[d, bend left, "\rho"] \arrow[d, phantom, "\overset{E}{\Rrightarrow}"] \\
te
\end{tikzcd}$$

but 3 are derivable…

Algebras

Given a 2-LW $\mathcal{T}$, we write $\mathsf{Alg}(\mathcal{T})$ for the 2-category of algebras of $\mathcal{T}$.

Theorem
A morphism $\mathsf{F} : \mathcal{T} \to \mathcal{T}'$ *of theories is a biequivalence if and only if the functor* $\mathsf{Alg}(F) : \mathsf{Alg}(\mathcal{T}') \to \mathsf{Alg}(\mathcal{T})$ *induced by precomposition is a biequivalence.*

Algebras

Given a 2-LW $\mathcal{T}$, we write $\mathsf{Alg}(\mathcal{T})$ for the 2-category of algebras of $\mathcal{T}$.

Theorem
A morphism $\mathsf{F} : \mathcal{T} \to \mathcal{T}'$ *of theories is a biequivalence if and only if the functor* $\mathsf{Alg}(F) : \mathsf{Alg}(\mathcal{T}') \to \mathsf{Alg}(\mathcal{T})$ *induced by precomposition is a biequivalence.*

Given $\mathcal{W}$ **2**-rigid if we could show that the functor

$$\mathcal{T} \to \mathcal{T}/\mathcal{W}$$

is a biequivalence, we would deduce that

$$\mathsf{Alg}(\mathcal{T}) \to \mathsf{Alg}(\mathcal{T}/\mathcal{W})$$

is a biequivalence... but this is not the case!

Local equivalences vs biequivalences

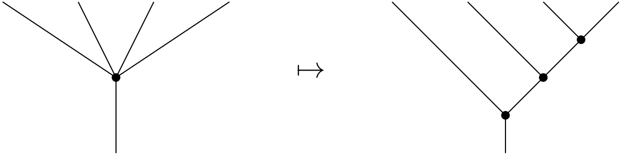
With $\mathcal{W}$ = all 2-cells, the functor

$$\mathsf{Mon} \to \mathsf{Mon}/\mathcal{W}$$

is a local equivalence (an equivalence on homs), there is a natural operation

$$\mathsf{Mon}/\mathcal{W} \to \mathsf{Mon}$$

but this is only a pseudofunctor:



A conjecture

Conjecture
When $\mathcal{W}$ is $\mathbf{2}$-rigid, the canonical 2-functor $\mathsf{Alg}(\mathcal{T}/\mathcal{W}) \to \mathsf{Alg}(\mathcal{T})$ *has a left adjoint such that the components of the unit are equivalences.*

The case of symmetric monoidal categories

The theory of commutative monoids is

$$\left\langle \begin{array}{r} m : 2 \\ e : 0 \end{array} \;\middle|\; \begin{array}{rr} \alpha : & m(m(x_1, x_2), x_3) \Rightarrow m(x_1, m(x_2, x_3)) \\ \lambda : & m(e, x_1) \Rightarrow x \\ \rho : & m(x_1, e) \Rightarrow x \\ \gamma : & m(x_1, x_2) \Rightarrow m(x_2, x_1) \end{array} \;\middle|\; \begin{tikzcd} & m(x_2, x_1) \arrow[dr, Rightarrow, "\gamma"] & \\ m(x_1, x_2) \arrow[ur, Rightarrow, "\gamma"] \arrow[rr, bend right, equal] \arrow[rr, phantom, "G\Lleftarrow" below] & & m(x_1, x_2) \end{tikzcd} \quad \cdots \right\rangle$$

- if we take $\mathcal{W}$ generated by α, λ, ρ, we are convergent:
  every symmetric monoidal category is equivalent to a strict one
- but we can do more!

The case of symmetric monoidal categories

The theory of commutative monoids is

$$\left\langle \begin{array}{r} m : 2 \\ e : 0 \end{array} \;\middle|\; \begin{array}{rr} \alpha : & m(m(x_1,x_2),x_3) \Rightarrow m(x_1,m(x_2,x_3)) \\ \lambda : & m(e,x_1) \Rightarrow x \\ \rho : & m(x_1,e) \Rightarrow x \\ \gamma : & m(x_1,x_2) \Rightarrow m(x_2,x_1) \end{array} \;\middle|\; \begin{tikzcd} & m(x_2,x_1) \arrow[dr, Rightarrow, "\gamma"] & \\ m(x_1,x_2) \arrow[ur, Rightarrow, "\gamma"] \arrow[rr, equal, bend right=40] \arrow[rr, phantom, "G\Downarrow"] & & m(x_1,x_2) \end{tikzcd} \quad \ldots \right\rangle$$

- it is locally confluent (critical branchings can be closed)
- it is not terminating otherwise we could show “full coherence” including

$$\begin{tikzcd} X \otimes X \arrow[r, bend left, "\gamma_{X,X}"] \arrow[r, bend right, "\mathrm{id}_{X\otimes X}"'] & X \otimes X \end{tikzcd}$$

The case of symmetric monoidal categories

The theory of commutative monoids is

$$\left\langle \begin{array}{r|rl|c} \begin{array}{r} m : 2 \\ e : 0 \end{array} & \begin{array}{rr} \alpha : & m(m(x_1, x_2), x_3) \Rightarrow m(x_1, m(x_2, x_3)) \\ \lambda : & m(e, x_1) \Rightarrow x \\ \rho : & m(x_1, e) \Rightarrow x \\ \gamma : & m(x_1, x_2) \Rightarrow m(x_2, x_1) \end{array} & & \begin{tikzcd} & m(x_2, x_1) \arrow[dr, Rightarrow, "\gamma"] & \\ m(x_1, x_2) \arrow[ur, Rightarrow, "\gamma"] \arrow[rr, bend right, equal] \arrow[rr, phantom, "G\Ddownarrow"] & & m(x_1, x_2) \end{tikzcd} \quad \ldots \end{array} \right\rangle$$

- restricting to affine terms (without repeated variables is not enough):

$$m(x_1, x_2) \xRightarrow{\gamma(x_1, x_2)} m(x_2, x_1) \xRightarrow{\gamma(x_2, x_1)} m(x_1, x_2)$$

- but we don't need both $m(x_1, x_2) \Rightarrow m(x_2, x_1)$ and $m(x_2, x_1) \Rightarrow m(x_1, x_2)$

The case of symmetric monoidal categories

The theory of commutative monoids is

$$\left\langle \begin{array}{r} m : 2 \\ e : 0 \end{array} \;\middle|\; \begin{array}{lrl} \alpha : & m(m(x_1,x_2),x_3) & \Rightarrow m(x_1,m(x_2,x_3)) \\ \lambda : & m(e,x_1) & \Rightarrow x \\ \rho : & m(x_1,e) & \Rightarrow x \\ \gamma : & m(x_1,x_2) & \Rightarrow m(x_2,x_1) \end{array} \;\middle|\; \right.$$

$$\begin{tikzcd} & m(x_2,x_1) \arrow[dr, Rightarrow, "\gamma"] & \\ m(x_1,x_2) \arrow[ur, Rightarrow, "\gamma"] \arrow[rr, bend right, equal] \arrow[rr, phantom, "G\Downarrow"] & & m(x_1,x_2) \end{tikzcd}$$

$$\left. \ldots \right\rangle$$

- if we only keep morphisms “sorting variables”, we are almost terminating excepting for situations such as $m(e,e) \Rightarrow m(e,e)$ which can be removed:

$$\begin{tikzcd} m(e,e) \arrow[rr, Rightarrow] \arrow[dr, Rightarrow] & & m(e,e) \arrow[dl, Rightarrow] \\ & e & \end{tikzcd}$$

The case of symmetric monoidal categories

The theory of commutative monoids is

$$\left\langle \begin{array}{r} m : 2 \\ e : 0 \end{array} \;\middle|\; \begin{array}{rr} \alpha : & m(m(x_1, x_2), x_3) \Rightarrow m(x_1, m(x_2, x_3)) \\ \lambda : & m(e, x_1) \Rightarrow x \\ \rho : & m(x_1, e) \Rightarrow x \\ \gamma : & m(x_1, x_2) \Rightarrow m(x_2, x_1) \end{array} \;\middle|\; \begin{tikzcd} & m(x_2, x_1) \arrow[dr, Rightarrow, "\gamma"] & \\ m(x_1, x_2) \arrow[ur, Rightarrow, "\gamma"] \arrow[rr, equal, bend right] \arrow[rr, phantom, "G\Downarrow"] & & m(x_1, x_2) \end{tikzcd} \quad \ldots \right\rangle$$

Theorem
In a symmetric monoidal category, every diagram whose source is a tensor product of distinct objects commutes.

Part III

Conclusion

Rigidity!

A quotient of (2-)category by a subgroupoid $\mathcal{W}$ is **coherent** when $\mathcal{W}$ is **rigid**.

This is the case when $\mathcal{W}$ is generated by a **convergent rewriting system**.

This also explains situations such as coherence for rig categories:

$$\delta_{x,y,z} : x \otimes (y \oplus z) \to (x \otimes y) \oplus (x \otimes z)$$

$$\delta'_{x,y,z} : (x \oplus y) \otimes z \to (x \otimes z) \oplus (y \otimes z)$$

$$\begin{tikzcd}
& (a+b)(c+d) \arrow[ld] \arrow[rd] & \\
a(c+d)+b(c+d) \arrow[d] & & (a+b)c+(a+b)d \arrow[d] \\
ac+ad+bc+bd \arrow[rr, "\sim"'] & & ac+bc+ad+bd
\end{tikzcd}$$

Thanks!

Questions?