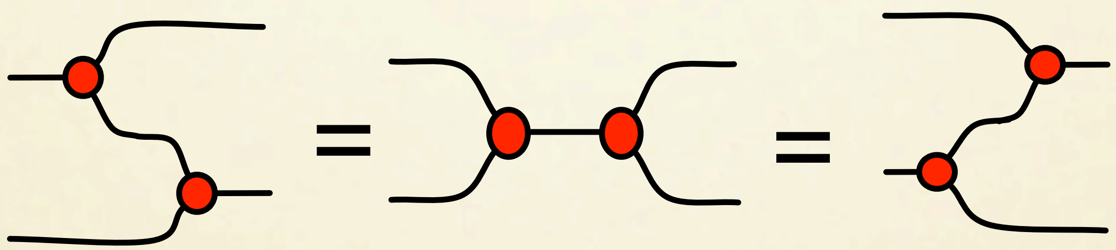
Rewriting with Frobenius

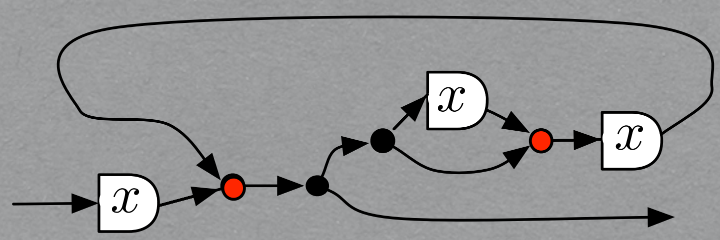


F. Bonchi F. Gadducci A. Kissinger P. Sobocinski **F. Zanasi**

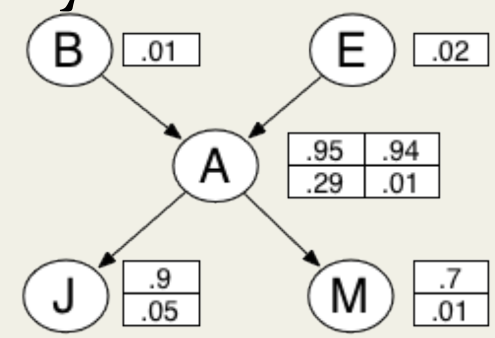
Logic in Computer Science 2018

Graphical Network Algebra

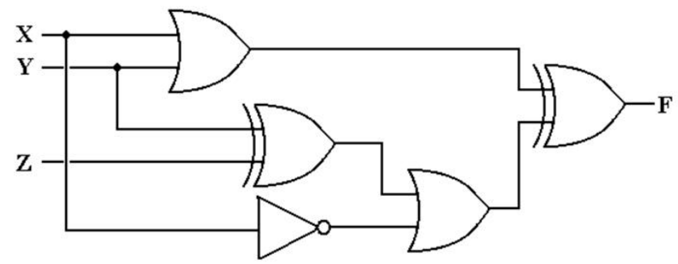
Signal Flow Graphs



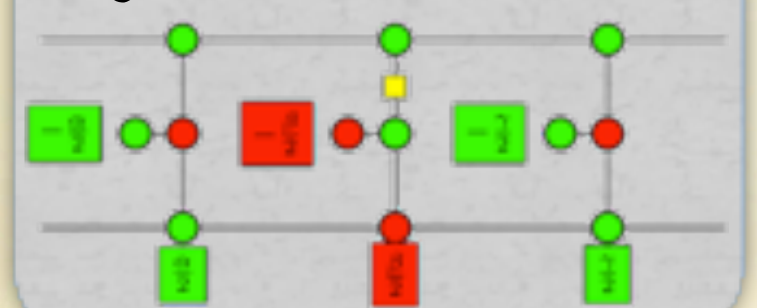
Bayesian Networks



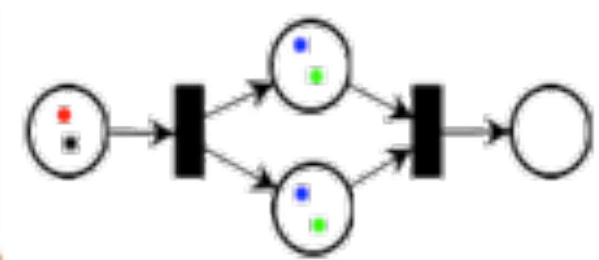
Digital Circuits



Quantum Processes



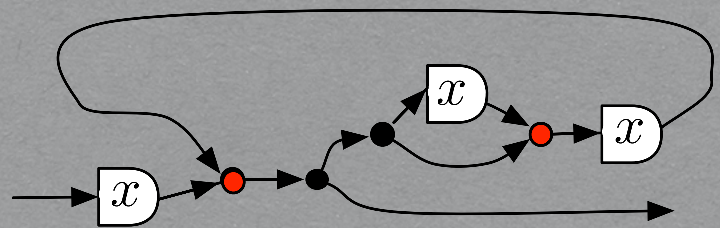
Petri Nets

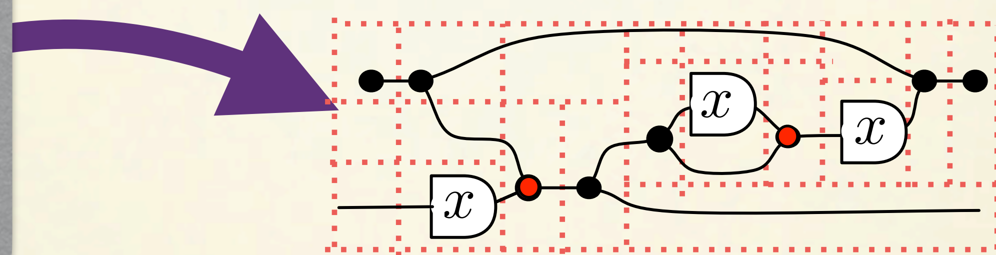


Categorical framework: **props** (SMCs with set of objects N).

Example

Signal Flow Graphs

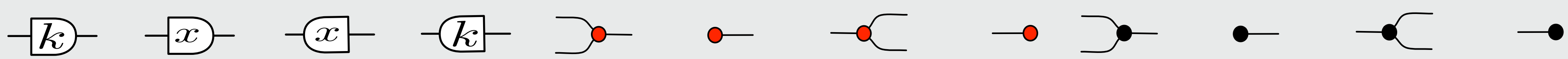




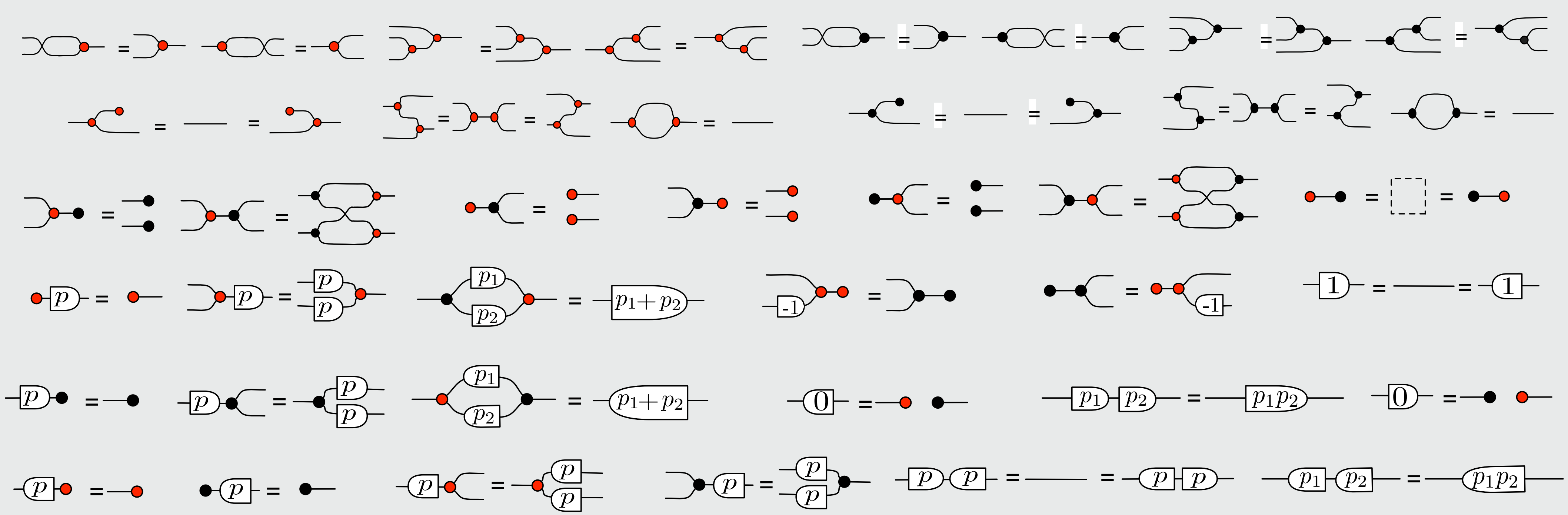
Props can be freely constructed starting from a signature and equations.

prop of signal flow graphs

Signature

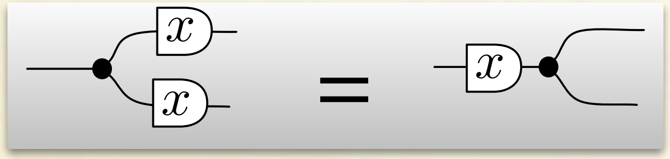


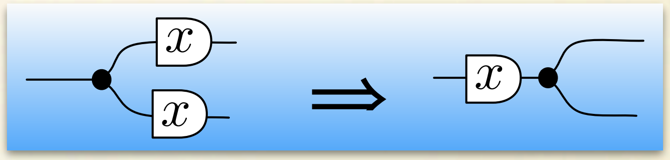
Equations



Rewriting in a prop

- This work focusses on **string diagram rewriting** in a prop.
  - Equations between diagrams are oriented as **rewrite rules**

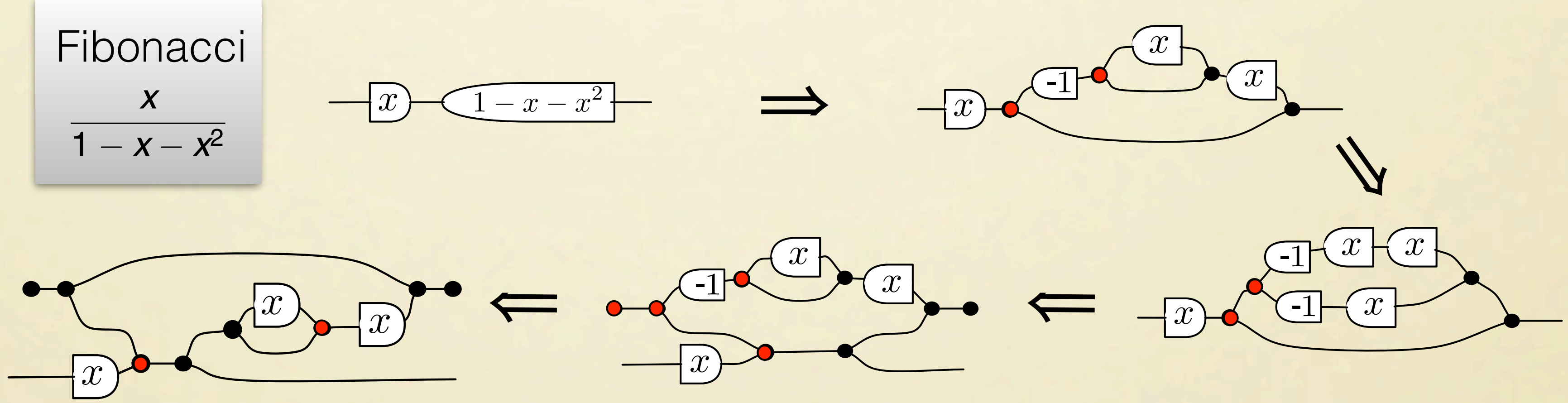




- Rewriting is useful to formalise **domain-specific procedures**
  - Example: realise a circuit specification into an executable signal flow graph

Fibonacci

$$\frac{x}{1-x-x^2}$$

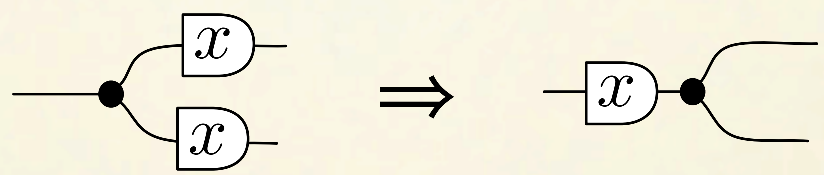


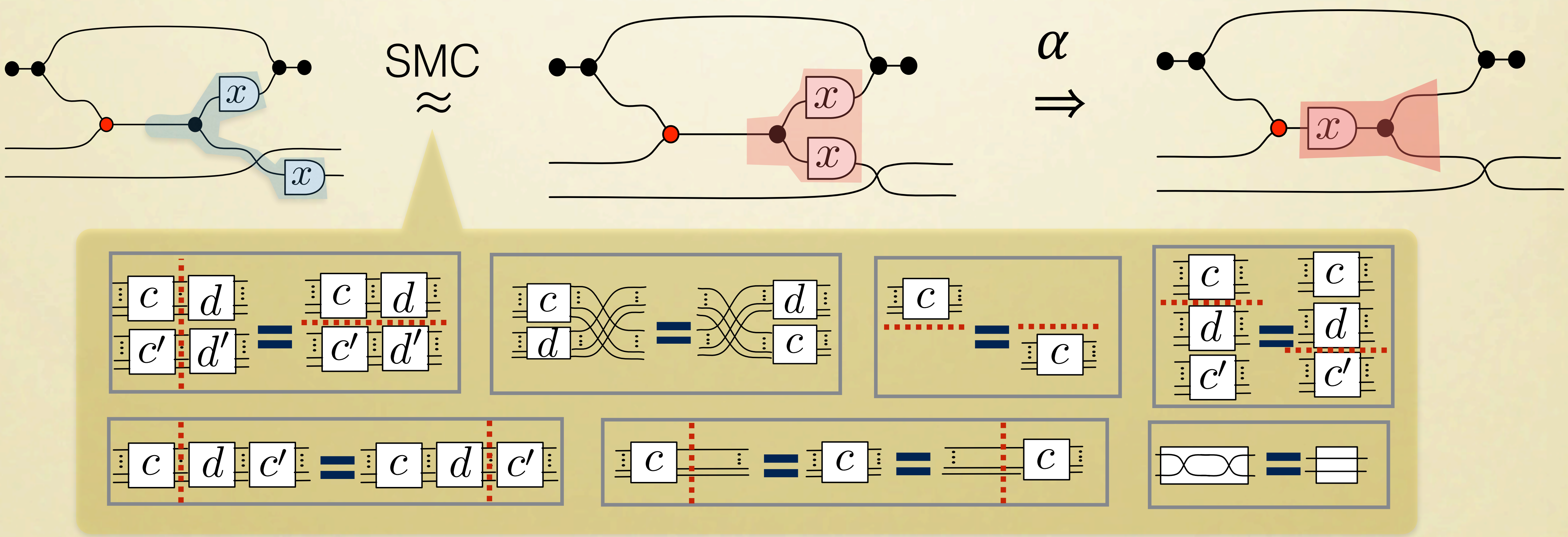
- Rewriting is useful to prove properties of equational theories of diagrams, such as **completeness, termination, confluence, decidability.**

Rewriting in a prop is tricky

How does **matching** work for diagram rewriting?

α :





Rewriting happens **modulo the laws of Symmetric Monoidal Categories**

Implementing rewriting modulo SMC

How do we implement this style of reasoning in a data structure which ``absorbs'' the structural rules of SMCs?

General recipe

String diagram	$\longmapsto$	Some flavour of graph
Diagram rewriting	$\longmapsto$	Some flavour of double-pushout (DPO) graph rewriting

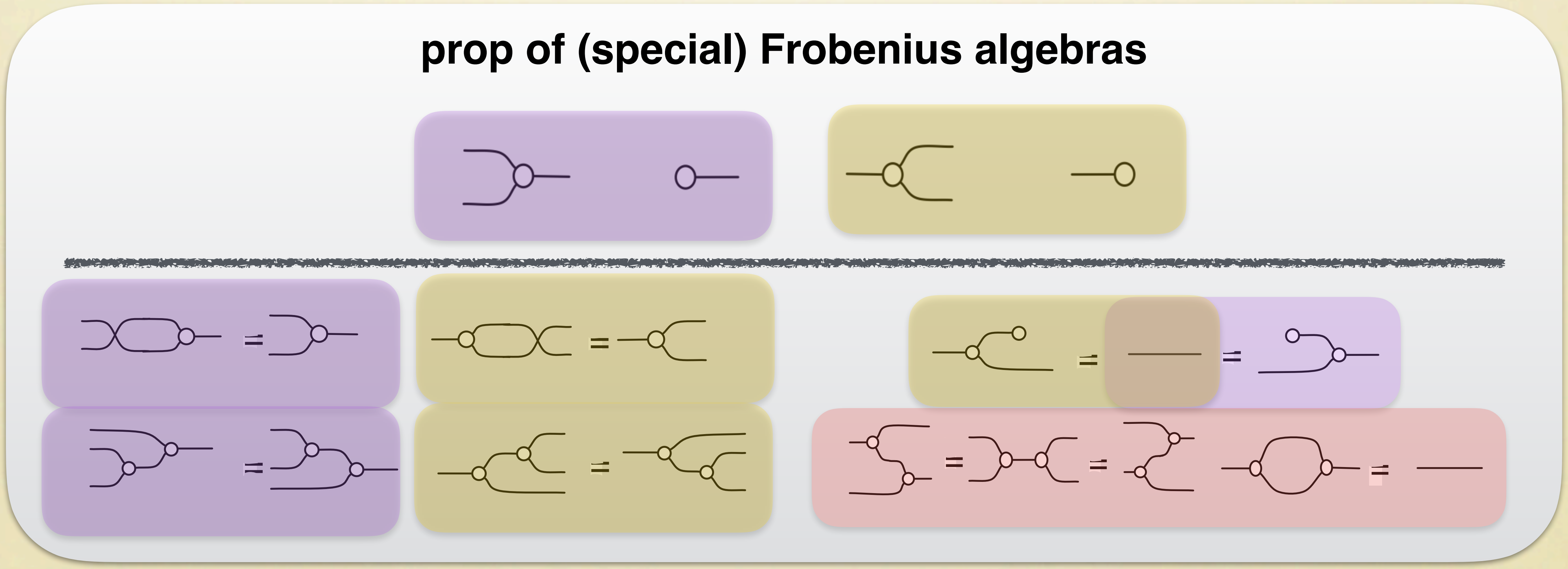
In previous work (LICS'16):

Diagram rewriting modulo SMC	$\longmapsto$	``Convex''DPO graph rewriting

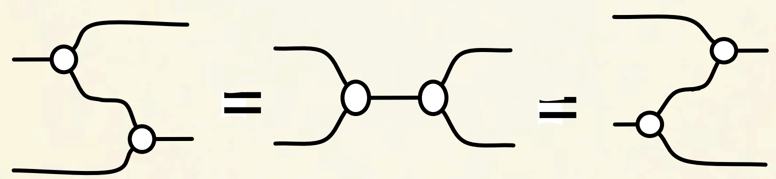
One step further: Frobenius algebras

This work: go beyond symmetric monoidal and study rewriting modulo more sophisticated algebraic structure.

We focus on **Frobenius algebra structures**.



Frobenius algebras are ubiquitous



- Diagrammatic calculi for **circuits** (like signal flow graphs, digital circuits, …): elementary connectors for ``sum'' and ``copy'' form Frobenius algebras.

- In the **ZX-calculus**: two Frobenius algebras, one for each interacting quantum observable.

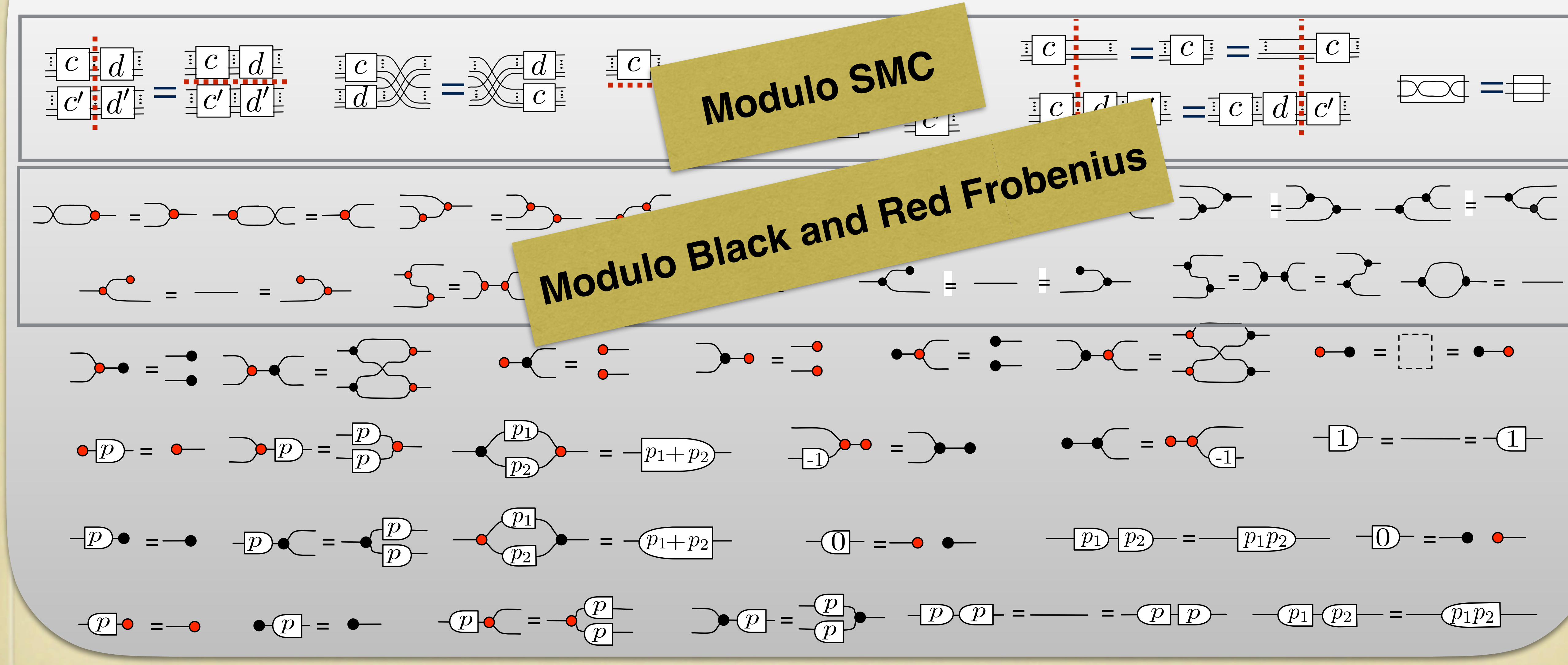
- **Hypergraph categories**: each object carries a Frobenius algebra.

- More: **Cospans, Cartesian Bicategories,** ….

Frobenius algebras are ubiquitous

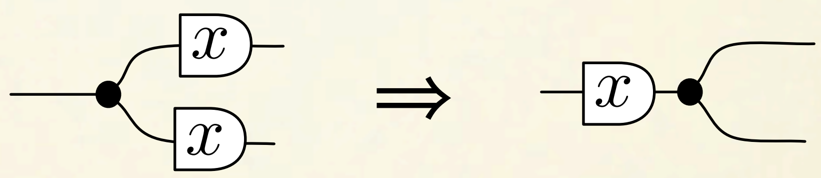
By considering the Frobenius algebras as **structural**, we drastically simplify the rewriting theory of all these calculi.

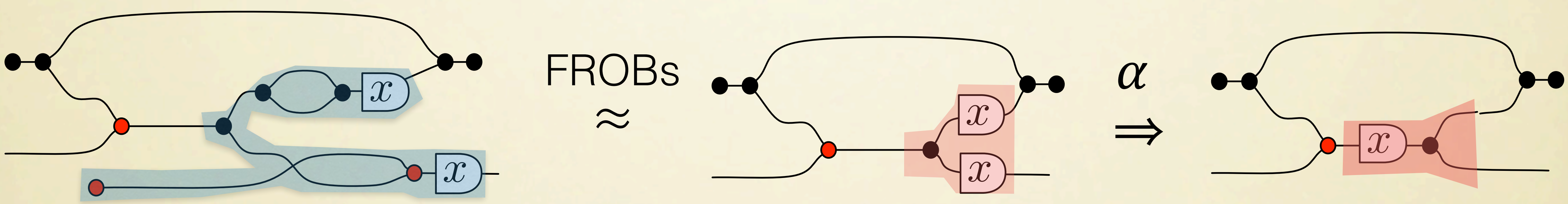
Example: the prop of signal flow graphs



Rewriting modulo Frobenius(es)

α :





How do we implement that?

Rewriting modulo Frobenius(es)

Goal of this work: identify a DPO rewriting interpretation which is **sound and complete** for reasoning modulo Frobenius structures

Strategy:

Diagram rewriting modulo Frobenius algebras in a prop $\overset{\text{Step 1}}{\longmapsto}$ Diagram rewriting modulo Frobenius algebras in a **multi-sorted** prop $\overset{\text{Step 2}}{\longmapsto}$ Some flavour of DPO graph rewriting

Props are **single-sorted** algebraic theories.

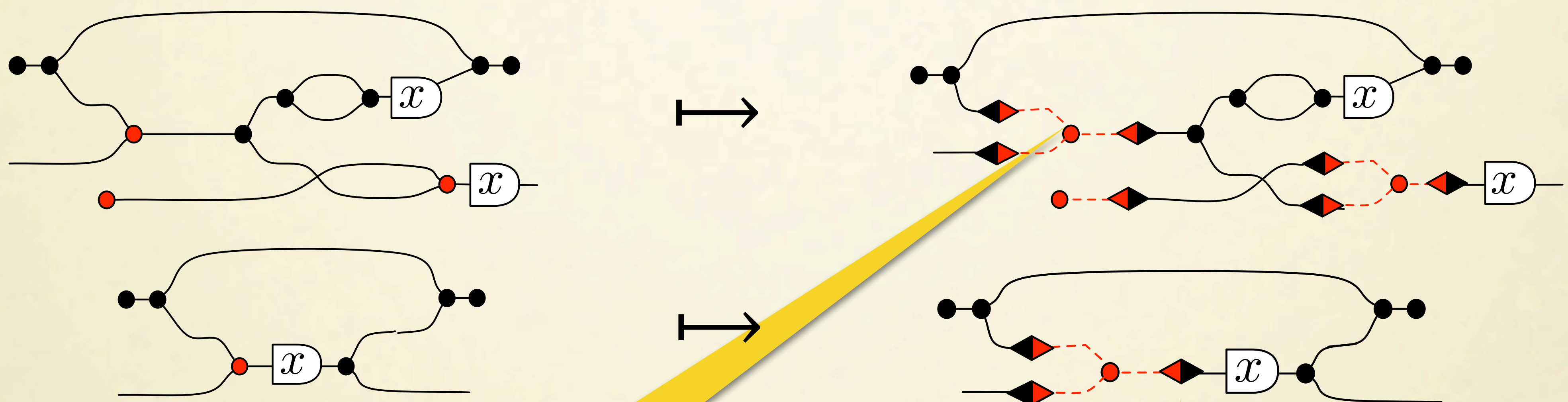
We need **multiple** sorts: one for each Frobenius algebra.

Graph interpretation: one sort of nodes = one Frobenius algebra.

Step 1: from props to multi-sorted props

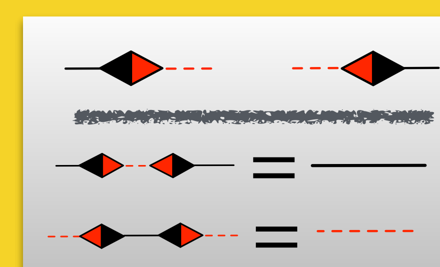
Prop

Multi-sorted prop (black & red)



Each Frobenius algebra is now on a different sort.

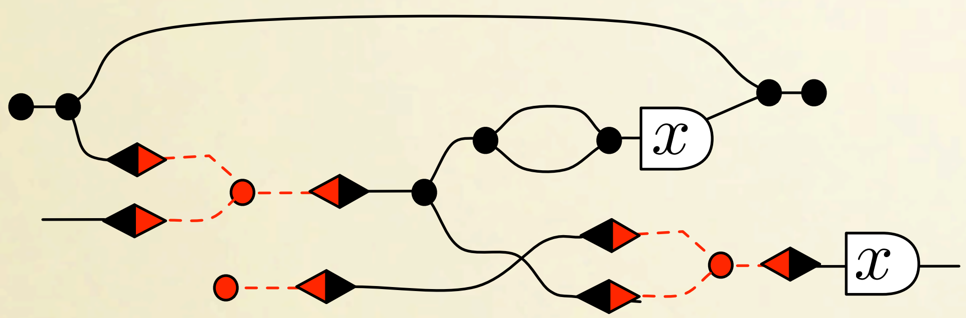
We introduce **``sort-switch" connectors** to maintain the diagram shape well-typed.



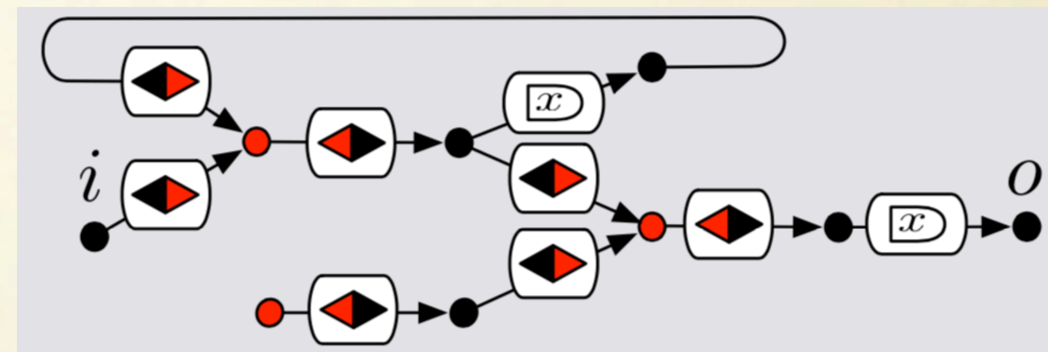
Proposition This is an *equivalence* of categories.

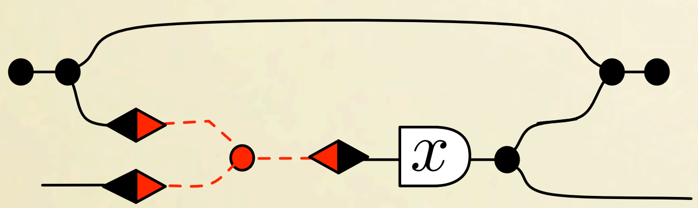
Step 2: from multi-sorted props to graphs

Multi-sorted prop (black & red) | **Hypergraphs (with interfaces)**

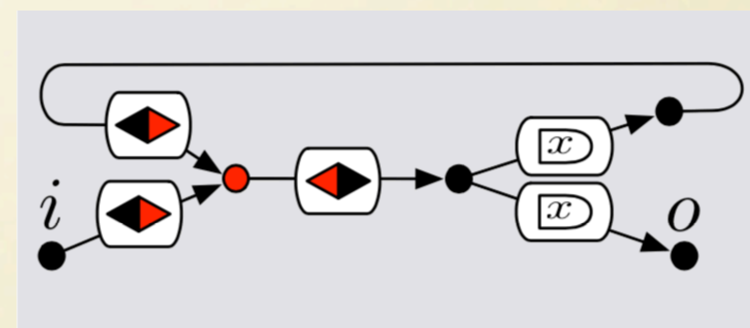


$\longmapsto$





$\longmapsto$



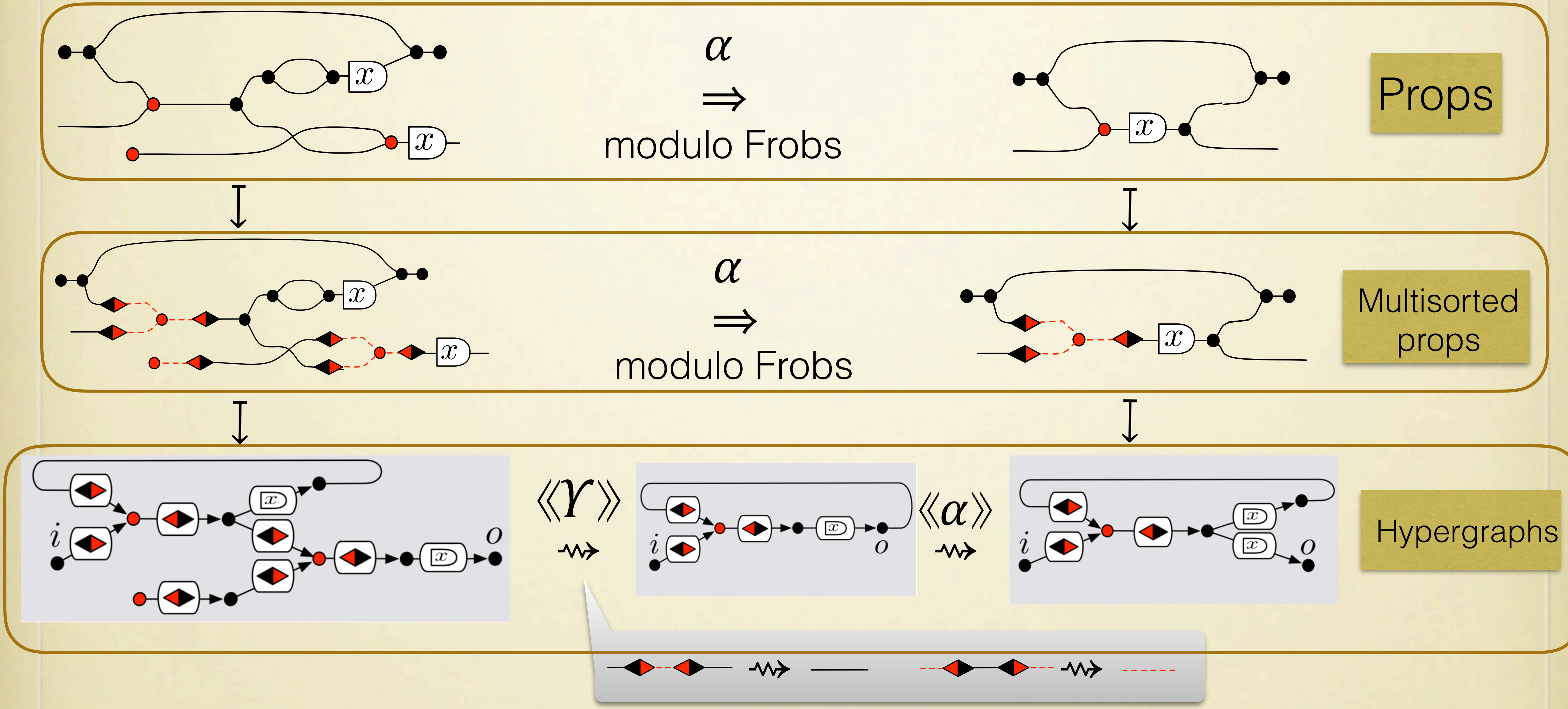
Theorem

Rewriting in *C*-sorted props with a Frobenius algebra on each sort in *C*.

$\Longleftrightarrow$

DPO rewriting of hypergraphs with *C*-sorted nodes.

Putting everything together



Main Theorem

Rewriting props with C Frobenius algebras. ⟺ DPO rewriting of hypergraphs with C-sorted nodes, in Υ-normal form.

Conclusions

Main result

a sound and complete DPO rewriting implementation for reasoning modulo **multiple** Frobenius structures.

Conceptual core

We built on an equivalence between props and coloured props, in which Frobenius algebras are changed of sort.

This is part of a broader picture (an adjunction) that remains to be investigated.

Applications

Graphical reasoning in theories with multiple Frobenius algebras is drastically simplified: lots of equations become structural.

In the paper: the first terminating rewriting strategy for the theory of *interacting bialgebras* (aka the phase-free ZX-calculus).