DisCoPy: Monoidal Categories in Python

Giovanni de Felice, Alexis Toumi, Bob Coecke

Department of Computer Science, University of Oxford.
Cambridge Quantum Computing Ltd.

July 8th, ACT 2020

Outline

0. What is a diagram?
1. Free categories, functors and their Python implementation.
2. Pre-monoidal categories, drawing and reading diagrams.
3. Rewriting modulo interchangers, normal forms for diagrams.
4. Applications: natural language and quantum circuits.
5. Demo time!

What is a diagram?

1. Intuitively: a string diagram is a way of describing systems and processes, e.g. a cooking recipe, a quantum protocol, etc.
2. Algebraically: a string diagram is a morphism in the free monoidal category $\mathbf{MC}(\Sigma)$ generated by a signature $\Sigma_1 \rightrightarrows \Sigma_0^\star$.
3. Geometrically: a string diagram is 1d cell complex embedded in the plain, labeled with Σ.
4. Combinatorially: a string diagram is given by $\mathtt{dom}, \mathtt{cod} \in \Sigma_0^\star$, $\mathtt{boxes} \in \Sigma_1^\star$ and $\mathtt{offsets} \in \mathbb{N}^\star$ quotiented by interchangers.

Free categories, functors and their Python implementation.

Definition

Given a (simple) signature $\Sigma_1 \rightrightarrows \Sigma_0$, the free category $\mathbf{C}(\Sigma)$ has objects Σ_0 and arrows $f : s \to t$ given by lists $f = f_1 \ldots f_n \in \Sigma_1^n$ such that $\mathtt{dom}(f_1) = s, \mathtt{cod}(f_n) = t$ and $\mathtt{cod}(f_i) = \mathtt{dom}(f_{i+1})$ for all $i \leq n$. Identity arrows are given by the empty list and composition is given by list contatenation.

Proposition

A functor $F : \mathbf{C}(\boldsymbol{\Sigma}) \to \mathbf{D}$ is uniquely defined by a morphism of signatures $\Sigma \to U(\mathbf{D})$.

Implementation

class `Ob(name)`

class `Arrow(dom, cod, boxes)`

class `Box(name, dom, cod)`

class `Functor(ob, ar, ob_factory, ar_factory)`

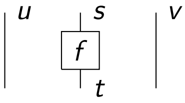
Pre-monoidal categories, drawing and reading diagrams.

Definition

A (strict) pre-monoidal category is a category $\mathbf{C}$ with a unital, associative functor $\boxtimes : \mathbf{C} \mathbin{\square} \mathbf{C} \to \mathbf{C}$ where $\square$ is given by the pushout of $\mathbf{C}_0 \times \mathbf{D} \leftarrow \mathbf{C}_0 \times \mathbf{D}_0 \to \mathbf{C} \times \mathbf{D}_0$ in $\mathbf{Cat}$.

Proposition

Given a (monoidal) signature $\Sigma_1 \rightrightarrows \Sigma_0^\star$, the free pre-monoidal category $\mathbf{PMC}(\Sigma)$ is given by the free category $\mathbf{C}(L(\Sigma))$ for the (simple) signature of layers $L(\Sigma) : \Sigma_0^\star \times \Sigma_1 \times \Sigma_0^\star \rightrightarrows \Sigma_0^\star$.

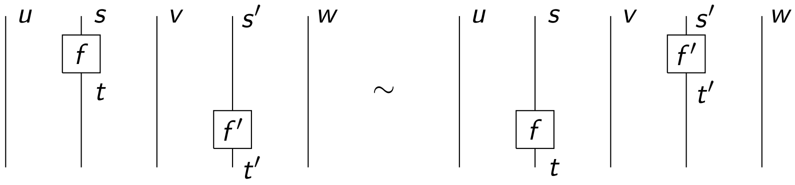


A layer $(u, f, v) \in \Sigma_0^\star \times \Sigma_1 \times \Sigma_0^\star$.

Rewriting modulo interchangers, normal forms for diagrams

Proposition

Given a (monoidal) signature $\Sigma_1 \rightrightarrows \Sigma_0^\star$, the free monoidal category $\mathbf{MC}(\Sigma)$ *is given by* $\mathbf{PMC}(\Sigma)/\mathcal{I}$ *for* $\mathcal{I}$ *the interchanger relation:*



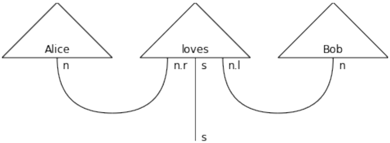
where $u, v, w \in \Sigma_0^\star$ *and* $s \xrightarrow{f} t, s' \xrightarrow{f'} t' \in \Sigma_1$.

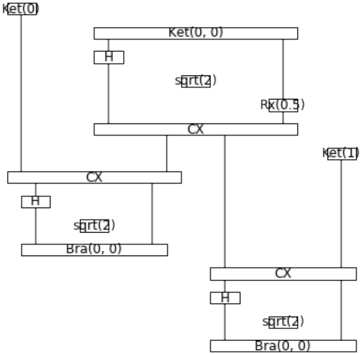
Theorem

The word problem for monoidal categories is decidable in polynomial time.[1]

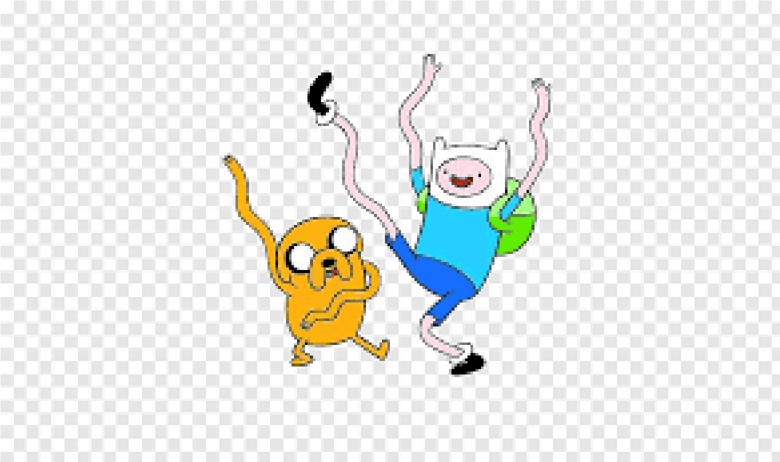
[1] Antonin Delpeuch, Jamie Vicary: *Normalization for planar string diagrams and a quadratic equivalence algorithm* (arXiv:1804.07832)

Applications: natural language and quantum circuits.





Demo time!



https://github.com/oxford-quantum-group/discopy