The Zeta Calculus

Nicklas Botö Fabian Forslund

Chalmers University of Technology

We propose an extension of the λ-**calculus** that models the behaviour of observables in quantum theory, the ζ-calculus. We give its definition as an extension of the internal language of **symmetric monoidal closed categories**, building upon its definition with the **observable structures** of the **ZX-calculus**. This introduces one abstraction for each observable structure, something which allows the definition of a **non-linear type system**. We give the semantics for the language as **string diagrams** in the underlying category. Introducing a model of the language in the category of finite dimensional Hilbert spaces, with the observable structures Z and X, recovers a λ-calculus with semantics in the ZX-calculus. We show that this language is universal for quantum computation and provide some examples of programming in it. A novel feature of the language is the ability to define controlled unitaries by the **control of bases in sharing**, without the need for a predefined gate set or control structures.

Syntax

We define the syntax of the ζ-calculus $\mathcal{X}$ inductively over a set of basis symbols $\mathcal{B}$. For each basis symbol $\zeta \in \mathcal{B}$, we fix a set of symbols $\mathcal{U}_\zeta$ called its unbiased points.

$$\frac{}{x \in \mathcal{X}} \qquad \frac{\zeta \in \mathcal{B} \quad \alpha \in \mathcal{U}_\zeta \quad n \in \mathbb{Z}}{\zeta^\alpha_n \in \mathcal{X}} \qquad \frac{\zeta \in \mathcal{B} \quad \alpha \in \mathcal{U}_\zeta \quad M \in \mathcal{X}}{\zeta^\alpha x M \in \mathcal{X}}$$

$$\frac{M \in \mathcal{X} \quad N \in \mathcal{X}}{MN \in \mathcal{X}} \qquad \frac{M \in \mathcal{X} \quad N \in \mathcal{X}}{\langle M, N\rangle \in \mathcal{X}} \qquad \frac{\zeta \in \mathcal{B} \quad M \in \mathcal{X} \quad N \in \mathcal{X}}{\text{let } \langle x, y\rangle =_\zeta M \text{ in } N \in \mathcal{X}}$$

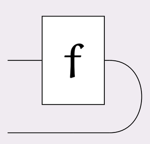
$$\frac{}{\star \in \mathcal{X}} \qquad \frac{M \in \mathcal{X} \quad N \in \mathcal{X}}{M \,;\, N \in \mathcal{X}}$$

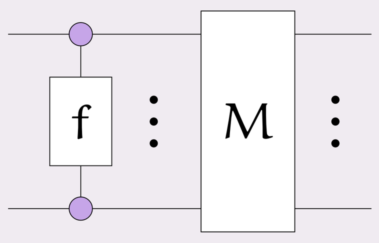
Embedding the linear λ-calculus

When an abstraction uses its variable exactly once, and applies no rotation, it doesn't utilise the observable it is introduced in. When this is the case, for a term $\zeta^0 x M$ where x occurs once in M, we write the term as $\lambda x M$.

Linking of variables

By defining the *linking function* $\ell_f :\equiv \lambda\langle x, y\rangle\, \eta^\dagger\langle f\,x, y\rangle$, we can act with an arbitrary function f on the arguments of a ζ-abstraction, linking these together, before being passed on to the body. The general form of this type of linking is written below.

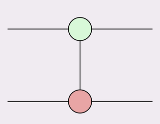
$\lambda\langle x, y\rangle\, \eta^\dagger\langle f\,x, y\rangle \rightsquigarrow$ 

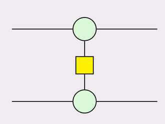
$\zeta x \zeta y\; \ell_f\langle x, y\rangle \,;\, M \rightsquigarrow$ 

Here, $\eta^\dagger :\equiv \zeta^0_{-2}$ is the induced compact structure of some observable. In a model where all the induced compact structures coincide we draw this as a cup or cap.

Controlled unitaries

When working in the model of finite-dimensional Hilbert spaces, using the set of observable structures $\mathcal{B} = \{\zeta, \xi\}$ of the ZX-calculus, we can apply ℓ to the identity function $I :\equiv \lambda x x$, to define the CNOT gate. Similarly, the controlled Z-gate can be defined by linking variables introduced in the Z-basis by Hadamard gate.

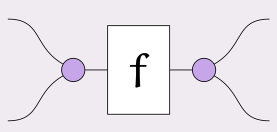
$\zeta c \xi t\; \ell_I\langle c, t\rangle \,;\, \langle c, t\rangle \rightsquigarrow$ 

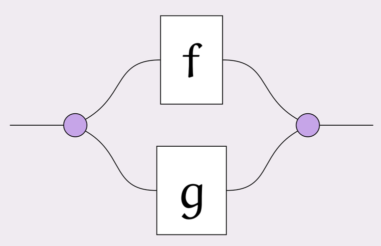
$\zeta c \zeta t\; \ell_H\langle c, t\rangle \,;\, \langle c, t\rangle \rightsquigarrow$ 

Thus, by gaining control of which basis a variable is shared in we are able to define these controlled unitaries without some specific control structures. Another construct that is able to be defined by linking variables together is the phase gadget.

Higher-order functions

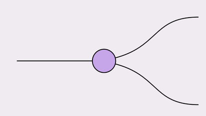
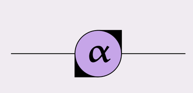
Since the ζ-calculus allows for higher-order functions and non-linearity, we are allowed to share functions. With this, we can construct terms which modify the behaviour of functions.

$\zeta f\langle f, f\rangle \rightsquigarrow$ 

$\lambda f \zeta g\; g \circ f \circ g \rightsquigarrow$ 

Semantics

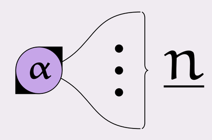
We give the semantics by mapping every derivable judgement to a string diagram in the underlying monoidal category, the interpretation is given by induction of the derivation. Each basis symbol in $\mathcal{B}$ is given an observable structure, defined by its comultiplication, counit, and phase group.

$\delta =$  $\epsilon =$ $\Lambda(\alpha) =$ 

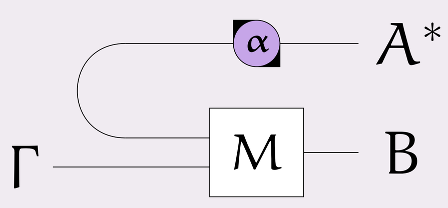
Type rules

$$\frac{}{x :_\zeta A \vdash x : A}$$

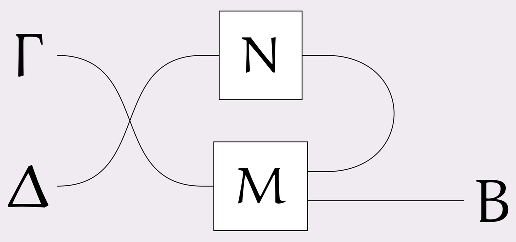
$$\frac{}{\vdash \zeta^\alpha_n : \underline{n}}$$



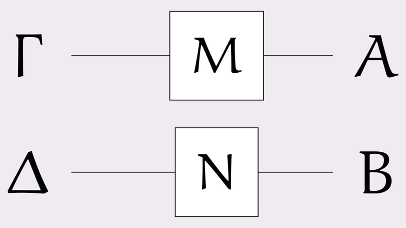
$$\frac{\Gamma, x :_\zeta A \vdash M : B}{\Gamma \vdash \zeta^\alpha x M : A \multimap B}$$



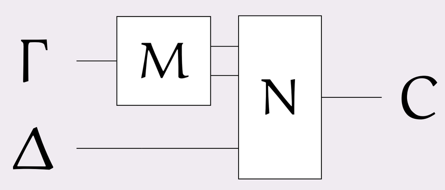
$$\frac{\Gamma \vdash M : A \multimap B \qquad \Delta \vdash N : A}{\Gamma, \Delta \vdash MN : B}$$



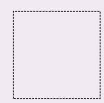
$$\frac{\Gamma \vdash M : A \qquad \Delta \vdash N : B}{\Gamma, \Delta \vdash \langle M, N\rangle : A \otimes B}$$



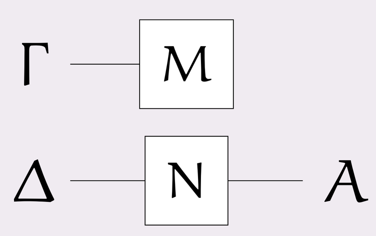
$$\frac{\Gamma \vdash M : A \otimes B \qquad \Delta, x :_\zeta A, y :_\zeta B \vdash N : C}{\Gamma, \Delta \vdash \text{let } \langle x, y\rangle =_\zeta M \text{ in } N : C}$$



$$\frac{}{\vdash \star : I}$$

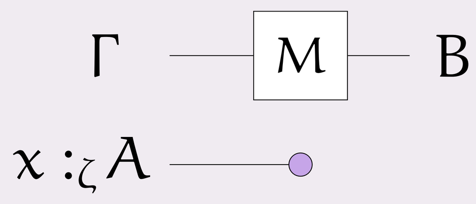


$$\frac{\Gamma \vdash M : I \qquad \Delta \vdash N : A}{\Gamma, \Delta \vdash M \,;\, N : A}$$

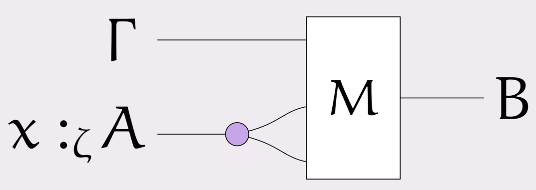


Structural rules

$$\frac{\Gamma \vdash M : B}{\Gamma, x :_\zeta A \vdash M : B}$$



$$\frac{\Gamma, x_1 :_\zeta A, x_2 :_\zeta A \vdash M : B}{\Gamma, x :_\zeta A \vdash M[x_1 := x, x_2 := x] : B}$$



$$\frac{\Gamma, x :_\zeta A, y :_\xi B, \Delta \vdash M : C}{\Gamma, y :_\xi B, x :_\zeta A, \Delta \vdash M : C}$$