An abstract model for Proto-Quipper-M extended with general recursion

Bert Lindenhovius  Michael Mislove  Vladimir Zamzhiev

Department of Computer Science
Tulane University
Proto-Quipper-M

- We will consider several variants of a functional programming language called Proto-Quipper-M.
- Language and model developed by Francisco Rios and Peter Selinger.
- Language is equipped with formal denotational and operational semantics.
- Primary application is in quantum computing, but the language can describe arbitrary string diagrams.
- Their model supports primitive recursion, but does not support general recursion.
Proto-Quipper-M is used to describe families of morphisms of an arbitrary, but fixed, symmetric monoidal category, which we denote $\mathbf{M}$.

**Example**

If $\mathbf{M} = \mathbf{FdCStar}$, the category of finite-dimensional $C^*$-algebras and completely positive maps, then a program in our language is a family of quantum circuits.
Circuit Model

Example

Shor’s algorithm for integer factorization may be seen as an infinite family of quantum circuits – each circuit is a procedure for factorizing an $n$–bit integer, for a fixed $n$.

Figure: Quantum Fourier Transform on $n$ qubits (subroutine in Shor’s algorithm).\(^1\)

\(^1\)Figure source: https://commons.wikimedia.org/w/index.php?curid=14545612
Syntax of Proto-Quipper-M

The type system is given by:

Types

\[ A, B ::= \alpha | 0 | A + B | I | A \otimes B | A \rightarrow B | !A | \text{Circ}(T, U) \]

Parameter types

\[ P, R ::= \alpha | 0 | P + R | I | P \otimes R | !A | \text{Circ}(T, U) \]

M-types

\[ T, U ::= \alpha | I | T \otimes U \]

The term language is given by:

Terms

\[ M, N ::= x | I | c | \text{let } x = M \text{ in } N \]
\[ \quad \quad | \Box_A M | \text{left}_{A,B} M | \text{right}_{A,B} M | \text{case } M \text{ of } \{ \text{left } x \rightarrow N | \text{right } y \rightarrow P \} \]
\[ \quad \quad | * | M; N | \langle M, N \rangle | \text{let } \langle x, y \rangle = M \text{ in } N | \lambda x^A.M | MN \]
\[ \quad \quad | \text{lift } M | \text{force } M | \text{box}_T M | \text{apply}(M, N) | (\tilde{I}, C, \tilde{l}) \]
Families Construction

The following construction is well-known.

**Definition**

Given a category $C$, we define a new category $\text{Fam}[C]$:

- Objects are pairs $(X, A)$ where $X$ is a discrete category and $A : X \to C$ is a functor.
- A morphism $(X, A) \to (Y, B)$ is a pair $(f, \phi)$ where $f : X \to Y$ is a functor and $\phi : A \to B \circ f$ is a natural transformation.
- Composition of morphisms is given by: $(g, \psi) \circ (f, \phi) = (g \circ f, \psi f \circ \phi)$.

**Remark**

$\text{Fam}[C]$ is the free coproduct completion of $C$ and as a result has all small coproducts.

**Proposition**

If $C$ is a symmetric monoidal closed and product-complete category, then $\text{Fam}[C]$ is a symmetric monoidal closed category.
Categorical Model

Definition

- A symmetric monoidal closed and product-complete category $\overline{M}$.
- A fully faithful strong monoidal embedding $M \to \overline{M}$.
- A symmetric monoidal closed category $\text{Fam}[\overline{M}]$ which we will refer to as $\text{Fam}$.
- A symmetric monoidal adjunction:

\[
\begin{array}{ccc}
\text{Set} & & \text{Fam} \\
\downarrow & \circ & \downarrow \\
\text{Fam}(I, -) & & \\
\end{array}
\]

Remark

Setting $\overline{M} := [M^{op}, \text{Set}]$ satisfies the first two requirements and can be done for any $M$. 


Categorical Model

Theorem (Rios & Selinger 2017)

Every categorical model of Proto-Quipper-M is computationally sound and adequate with respect to its operational semantics.

Question

Sam Staton: Why do you need the Fam construction for this?

Open Problem

Find a categorical model of Proto-Quipper-M which supports general recursion.
Our approach

• Describe an *abstract* categorical model for the same language.
• Describe an abstract categorical model for the language extended with recursion.

**Related work:** Rennela and Staton describe a different circuit description language where they also use enriched category theory.
A model of Intuitionistic Linear Logic (ILL) as described by Benton is given by the following data:

- A cartesian closed category $\mathbf{V}$.
- A symmetric monoidal closed category $\mathbf{L}$.
- A symmetric monoidal adjunction:

\[
\mathbf{V} \models \mathbf{L}
\]

\[
\begin{array}{c}
\mathbf{V} \\
\downarrow \quad \downarrow \\
\mathbf{L} \\
\end{array}
\]

\[
\mathbf{V} \xleftarrow{\perp} \mathbf{L}
\]

\[
\begin{array}{c}
\mathbf{V} \\
\downarrow \quad \downarrow \\
\mathbf{L} \\
\end{array}
\]

\[
\mathbf{V} \xleftarrow{\perp} \mathbf{L}
\]

\[
\begin{array}{c}
\mathbf{V} \\
\downarrow \quad \downarrow \\
\mathbf{L} \\
\end{array}
\]

\[
\mathbf{V} \xleftarrow{\perp} \mathbf{L}
\]

\[
\begin{array}{c}
\mathbf{V} \\
\downarrow \quad \downarrow \\
\mathbf{L} \\
\end{array}
\]

\[
\mathbf{V} \xleftarrow{\perp} \mathbf{L}
\]

\[
\begin{array}{c}
\mathbf{V} \\
\downarrow \quad \downarrow \\
\mathbf{L} \\
\end{array}
\]

\[
\mathbf{V} \xleftarrow{\perp} \mathbf{L}
\]

\[
\begin{array}{c}
\mathbf{V} \\
\downarrow \quad \downarrow \\
\mathbf{L} \\
\end{array}
\]

\[
\mathbf{V} \xleftarrow{\perp} \mathbf{L}
\]
Models of the Enriched Effect Calculus

A model of the Enriched Effect Calculus (EEC) is given by the following data:

• A cartesian closed category $\mathbf{V}$, enriched over itself.

• A $\mathbf{V}$-enriched category $\mathbf{L}$ with powers, copowers, finite products and finite coproducts.

• A $\mathbf{V}$-enriched adjunction:

\[ \begin{array}{ccc}
\mathbf{V} & \cong & \mathbf{L} \\
\downarrow & & \downarrow \\
\mathbf{V} & \cong & \mathbf{L}
\end{array} \]

Theorem

Every model of ILL with additives determines an EEC model.

---

An abstract model for Proto-Quipper-M

A model of Proto-Quipper-M is given by the following data:

1. A cartesian closed category $\mathcal{P}$ (the category of parameters) together with its self-enrichment $\mathcal{P}$, such that $\mathcal{P}$ has finite $\mathcal{P}$-coproducts.
2. A $\mathcal{P}$-symmetric monoidal category $\mathcal{M}$ with underlying category $\mathcal{M}$.
3. A $\mathcal{P}$-symmetric monoidal closed category $\mathcal{C}$ with underlying category $\mathcal{C}$ such that $\mathcal{C}$ has finite $\mathcal{P}$-coproducts.
4. A $\mathcal{P}$-strong symmetric monoidal functor $E : \mathcal{M} \to \mathcal{C}$.
5. A $\mathcal{P}$-symmetric monoidal adjunction: $\mathcal{P} \dashv \mathcal{C}$,

where $(\_ \odot I)$ denotes the $\mathcal{P}$-copower of the tensor unit in $\mathcal{C}$.

Remark: A model of PQM is essentially given by an enriched model of ILL.
Soundness

Theorem (Soundness)

Every abstract model of Proto-Quipper-M is computationally sound.
Concrete models of PQM

The original Proto-Quipper-M model is given by the model of ILL

\[ \text{Set} \xrightarrow{- \circ I} \text{Fam}[M] \xleftarrow{\bot} \text{Fam}[M](I, -) \]

Remark: When \( M = 1 \), the latter model degenerates to Set, which is a model of a simply-typed (non-linear) lambda calculus.
Concrete models of PQM

The original Proto-Quipper-M model is given by the model of ILL:

\[
\text{Set} \quad \perp \quad \text{Fam}[\overline{\mathcal{M}}] \\
\text{Fam}[\mathcal{M}](I, -) \quad \perp \quad \text{Fam}[\mathcal{M}](I, -)
\]

A simpler model for the same language is given by the model of ILL:

\[
\text{Set} \quad \perp \quad \overline{\mathcal{M}} \\
\mathcal{M}(I, -) \quad \perp \quad \mathcal{M}(I, -)
\]

where in both cases \( \overline{\mathcal{M}} = [\mathcal{M}^\text{op}, \text{Set}] \).

Remark

When \( \mathcal{M} = 1 \), the latter model degenerates to \( \text{Set} \) which is a model of a simply-typed (non-linear) lambda calculus.
Concrete models of the base language (contd.)

Fix an arbitrary symmetric monoidal category $M$. Equipping $M$ with the free DCPO-enrichment yields another concrete (order-enriched) Proto-Quipper-M model:

\[ M = [M^{\text{op}}, \text{DCPO}] \]

where $\overline{M} = [M^{\text{op}}, \text{DCPO}]$.

Remark

*The three concrete models of Proto-Quipper-M are EEC models whose underlying (unenriched) structure is a model of ILL.*
Abstract model with recursion?

Intuitionistic linear logics correspond to linear/non-linear lambda calculi under the Curry-Howard isomorphism.

**Theorem**

A categorical model of a linear/non-linear lambda calculus extended with recursion is given by a model of ILL:

\[ V \vdash L \]

where \( FG \) (or equivalently \( GF \)) is parametrically algebraically compact \(^2\).

\(^2\)Benton & Wadler. *Linear logic, monads and the lambda calculus*. LiCS’96.
Proto-Quipper-M extended with general recursion

Definition
A categorical model of PQM extended with general recursion is given by a model of
PQM, where in addition:

6. The comonad endofunctor:

\[ \mathcal{P} \xrightarrow{- \otimes I} \mathcal{C}, \]

is parametrically algebraically compact.

Moreover, if:

7. \( \mathcal{P} = \text{DCPO} \) and \( 0_{\mathcal{T},\mathcal{U}} \notin \text{Im}(E) \).

then we call this a \textit{computationally adequate} categorical model of PQM extended with
general recursion.
Recursion

Extend the syntax:

\[
\Phi, x :! A; \emptyset \vdash m : A \\
\Phi; \emptyset \vdash \text{rec } x^{!A} m : A
\]

Extend the operational semantics:

\[
(C, m[\text{lift rec } x^{!A} m/x]) \downarrow (C', v) \\
(C, \text{rec } x^{!A} m) \downarrow (C', v)
\]
Recursion (contd.)

Extend the denotational semantics: 

\[ \llbracket \Phi; \emptyset \vdash \text{rec } x^A \ m : A \rrbracket := \sigma_{[m]} \circ \gamma_{[\Phi]} \cdot \]

\[
\begin{align*}
\llbracket \Phi \rrbracket \otimes !\llbracket \Phi \rrbracket & \xleftarrow{\text{id} \otimes F\eta} \llbracket \Phi \rrbracket \otimes \llbracket \Phi \rrbracket \xrightarrow{\Delta} \llbracket \Phi \rrbracket \\
\llbracket \Phi \rrbracket \otimes !\llbracket \Omega \rrbracket & \xleftarrow{\omega_{[\Phi]}} \llbracket \Phi \rrbracket \\
\llbracket \Phi \rrbracket \otimes !\llbracket \Omega \rrbracket & \xrightarrow{\omega_{[\Phi]}} \llbracket \Omega \rrbracket \\
\llbracket \Phi \rrbracket \otimes !\llbracket A \rrbracket & \xrightarrow{\sigma_{[m]}} \llbracket m \rrbracket \\
\end{align*}
\]
Theorem (Soundess)

Every model of Proto-Quipper-M extended with recursion is computationally sound.

Theorem (Termination)

Consider a computationally adequate model of PQM extended with recursion. For any well-typed configuration \((C, m)\), if \(\llbracket (C, m) \rrbracket \neq 0\), then \((C, m) \Downarrow\). (Proof in progress).

Theorem (Adequacy)

Consider a computationally adequate model of PQM extended with recursion. For any well-typed configuration \((C, m)\), where \(m\) is a term of parameter type:

\[
\llbracket (C, m) \rrbracket \neq 0 \text{ iff } (C, m) \Downarrow
\]
Concrete model of Proto-Quipper-M extended with recursion

Let $M_*$ be the $\text{DCPO}_{\bot!}$-category obtained by freely adding a zero object to $M$ and $M_* = [M^\text{op}, \text{DCPO}_{\bot!}]$ be the associated enriched functor category.

Remark
If $M = 1$, then the above model degenerates to the left vertical adjunction, which is a model of a simply-typed lambda calculus with term-level recursion.
Fix an arbitrary symmetric monoidal category $M$.

Original Proto-Quipper-M model:

\[
\begin{align*}
\text{Set} & \xlongequal{\perp} \text{Fam}[M] \\
\text{Fam}[M](I, -) & \xlongequal{\perp} M(I, -)
\end{align*}
\]

Simpler model:

\[
\begin{align*}
\text{Set} & \xlongequal{\perp} \overline{M} \\
\overline{M}(I, -) & \xlongequal{\perp} M(I, -)
\end{align*}
\]

**Question**: What does the extra layer of abstraction provide?

**Answer**: A model of the language extended with dependent types.
Linear dependent types

Theorem
The category $\text{Fam}[M]$ is a model of dependently typed intuitionistic linear logic $^3$.

Conjecture

The symmetric monoidal adjunction:

$$\begin{array}{ccc}
\text{Set} & \xrightarrow{\perp} & \text{Fam}[M] \\
\downarrow & & \downarrow \\
\text{Fam}[M](I, -) & \xleftarrow{- \odot I} & 
\end{array}$$

is a model of Proto-Quipper-M extended with dependent types.

Remark

If $M = 1$, the above model degenerates to $\text{Fam}[M] = \text{Fam}[M^{\text{op}}, \text{Set}] \cong \text{Fam}[\text{Set}] \cong [2^{\text{op}}, \text{Set}]$, which is a closed comprehension category and thus a model of intuitionistic dependent type theory$^4$.


$^4$Bart Jacobs. *Categorical Logic and Type Theory*. 1999.
Abstract model with dependent types?

**Theorem**

A model of dependently typed intuitionistic linear logic is given by an indexed monoidal category with some additional structure (comprehension, strictness, ...).\(^5\)

**Conjecture**

An abstract model of Proto-Quipper-M extended with dependent types is given by an enriched indexed monoidal category\(^6\) with some additional structure (comprehension, strictness, ...).

---


What about recursion and dependent types simultaneously?

- This is the most complicated case by far.

\[
\begin{align*}
\text{DCPO} & \perp ! \quad \text{CFam} \perp ! \mathcal{M}_* \\
L & \perp \quad \text{CFam} \perp ! \mathcal{M}_[M](I,-) \\
\text{DCPO} & \perp \quad \text{CFam}[M] \\
L & \perp \quad \text{CFam}[M](I,-) \\
U & \perp \quad \text{CFam}[M] \\
\end{align*}
\]

Remark

If \( M = 1 \), then the model collapses to a model which is very similar to Palmgren and Stoltenberg-Hansen’s model of partial intuitionistic dependent type theory \(^7\).

Conjecture

An abstract model of Proto-Quipper-M extended with recursion and dependent types is given by an enriched indexed monoidal category with some additional structure (comprehension, strictness, ...) and suitable algebraic compactness conditions on the underlying adjoint functors.
Conclusion

• One can construct a model of PQM by categorically enriching certain denotational models.

• We described a sound abstract model for PQM.

• We described a sound and computationally adequate abstract model for PQM with general recursion.

• Systematic construction for concrete models that works for any circuit (string diagram) model described by a symmetric monoidal category.

• We have conjectured what possible models that support dependent types should look like.
Thank you for your attention!