Towards semantics for fault-tolerant quantum compilation

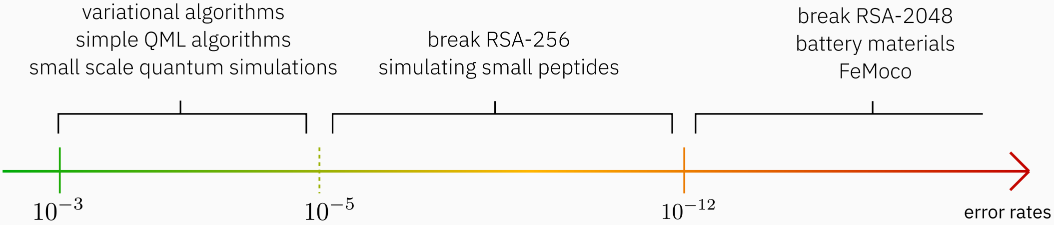
Robert I. Booth

RubberDUQ workshop, 16 September 2026

UNIVERSITY OF OXFORD

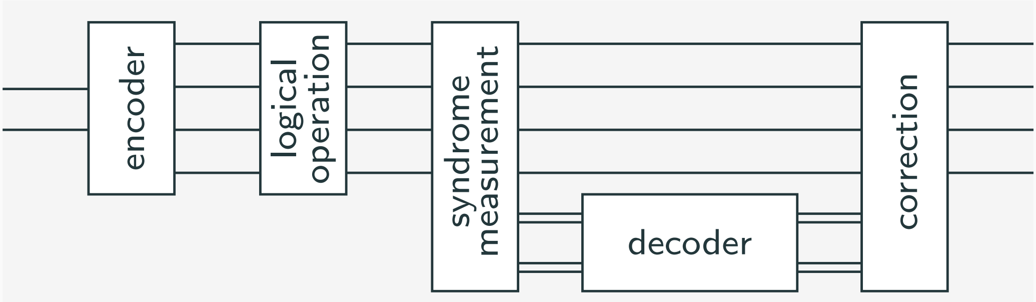
Noise in quantum hardware

error probability:	
transistor gates	qubits
0.000000000000001	0.001



Credit: Tamas Noszko, Adithya Sireesh

Fault-tolerant quantum computing



The fault-tolerant compilation problem

There are really two new correctness criteria here:

The fault-tolerant compilation problem

There are really two new correctness criteria here:

1. logical correctness, i.e. the compiled program actually manipulates the encoded data in the way we expect;

The fault-tolerant compilation problem

There are really two new correctness criteria here:

1. logical correctness, i.e. the compiled program actually manipulates the encoded data in the way we expect;
2. fault tolerance, i.e. the compiled program actually deals with noise correctly.

Stabiliser Programming Language

More pragmatically, we want to be able to reason about a simple programming language for stabiliser quantum mechanics:

$$c, d \ ::= c \mathbin{;} d \mid \overbrace{\textbf{init}\ \underline{\mathrm{x}} \mid \underline{\mathbf{y}} := f(\underline{\mathbf{x}}) \mid \textbf{disc}\ \underline{\mathrm{x}}}^{\textit{classical}} \mid \overbrace{\textbf{qinit}\ \underline{\mathrm{x}} \mid \underline{\mathbf{x}} \mathrel{*{=}} U}^{\textit{Clifford}}$$

$$\mid \underbrace{\underline{\mathrm{y}} := \textbf{meas}\ \underline{\mathrm{x}}}_{\textit{measurement}} \mid \underbrace{\textbf{ctrl}_P\ \underline{\mathrm{x}}\ \underline{\mathrm{y}}}_{\textit{classical control}}$$

Stabiliser Programming Language

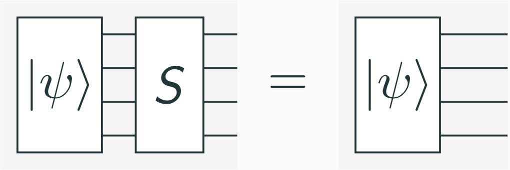
More pragmatically, we want to be able to reason about a simple programming language for stabiliser quantum mechanics:

$$c, d \;::=\; c \mathbin{;} d \mid \overbrace{\mathbf{init}\ \underline{x} \mid \underline{\mathbf{y}} := f(\underline{\mathbf{x}}) \mid \mathbf{disc}\ \underline{x}}^{\textit{classical}} \mid \overbrace{\mathbf{qinit}\ \underline{x} \mid \underline{\mathbf{x}} \mathrel{*}= U}^{\textit{Clifford}}$$
$$\mid \underbrace{\underline{y} := \mathbf{meas}\ \underline{x}}_{\textit{measurement}} \mid \underbrace{\mathbf{ctrl}_P\ \underline{x}\ \underline{y}}_{\textit{classical control}}$$

...explicitly exposing the code and error structure in order to be able to reason about fault-tolerance.

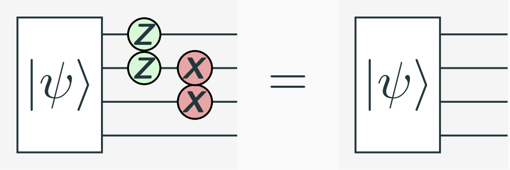
Stabiliser codes

Stabiliser codes are special QEC codes whose symmetries are *locally generated*:



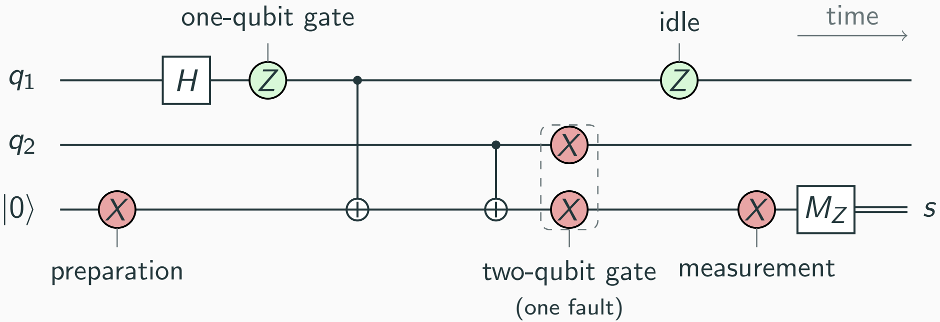
Stabiliser codes

Stabiliser codes are special QEC codes whose symmetries are *locally generated*:



Circuit-level noise

In a Pauli noise model, faults can occur throughout the circuit.



One faulty two-qubit gate can affect both output qubits.

An obstruction

Theorem 1 (Main result)**.** *Let $\Phi : \mathfrak{T}(\mathcal{H}_A) \to \mathfrak{T}(\mathcal{H}_B)$ be a multiqubit quantum channel. The following statements are equivalent:*

1. *Φ can be realized by a stabilizer circuit without classical control.*
2. *Φ sends pure stabilizer states to mixed stabilizer states.*

Vsevolod I. Yashin and Maria A. Elovenkova. **"Characterization of non-adaptive Clifford channels".** In: *Quantum Information Processing* 24.3 (), p. 99.

An obstruction

Theorem 1 (Main result)**.** *Let $\Phi : \mathfrak{T}(\mathcal{H}_A) \to \mathfrak{T}(\mathcal{H}_B)$ be a multiqubit quantum channel. The following statements are equivalent:*

1. *Φ can be realized by a stabilizer circuit without classical control.*
2. *Φ sends pure stabilizer states to mixed stabilizer states.*

Theorem 5 ([16])**.** *Let $\Phi : \mathfrak{T}(\mathcal{H}_A) \to \mathfrak{T}(\mathcal{H}_B)$ be a channel. The following statements are equivalent:*

1. *Φ is completely adaptive stabilizer preserving.*
2. *Choi state σ is a mixture of stabilizer states.*

Vsevolod I. Yashin and Maria A. Elovenkova. **"Characterization of non-adaptive Clifford channels".** In: *Quantum Information Processing* 24.3 (), p. 99.

Two strands progressing forwards

① How do we reason about QEC within these boundaries?

$\rightsquigarrow$ denotational semantics via symplectic geometry

② How to move beyond them?

$\rightsquigarrow$ quantum instruments as a computational effect

Denotational Semantics for Stabiliser Quantum Programs

Robert I. Booth
University of Oxford, United Kingdom

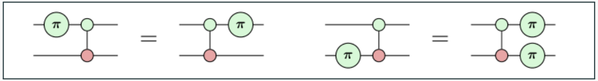
Cole Comfort
Université Paris-Saclay, CNRS, ENS Paris-Saclay, Inria, CentraleSupélec, Laboratoire Méthodes Formelles, France

Abstract

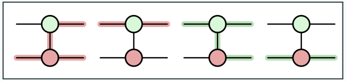
The stabiliser fragment of quantum theory is a foundational building block for quantum error correction, and hence for the fault-tolerant compilation of quantum programs. In this article, we develop a sound, universal, and complete denotational semantics for stabiliser operations, including measurement, classically controlled Pauli operators, and affine classical computation, thereby supporting an explicit treatment of quantum error-correcting codes. We interpret stabiliser operations as *affine relations* over finite fields, yielding a semantics that reflects the algebraic structure underlying stabiliser quantum error correction. Because stabiliser quantum mechanics has a well-behaved algebraic structure, our relational semantics is conceptually transparent and computationally tractable when compared to standard denotational models for general quantum programs. We demonstrate the resulting semantics by describing a small, low-level assembly language for stabiliser programs with fully abstract denotational semantics.

Pauli pushing in all its forms

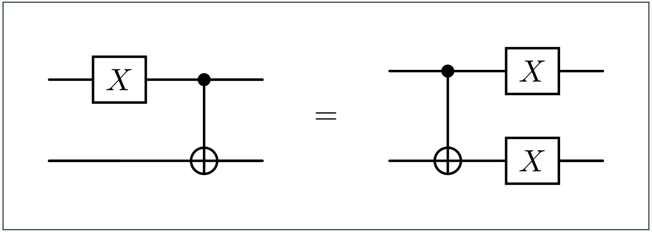
Pauli pushing



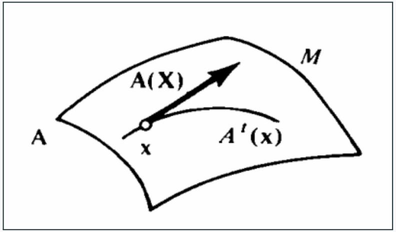
Pauli webs = stabiliser flows



Pauli propagation

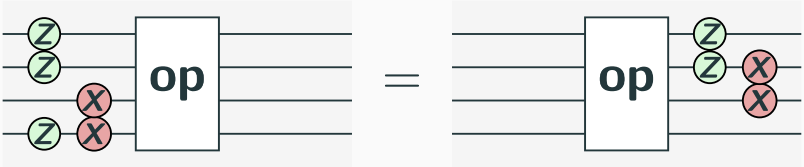


Lagrangian relations



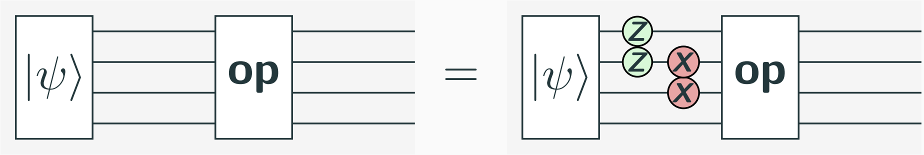
Stabiliser maps

We can define a subcategory of FdHilb of linear maps that push/propagate stabilisers:



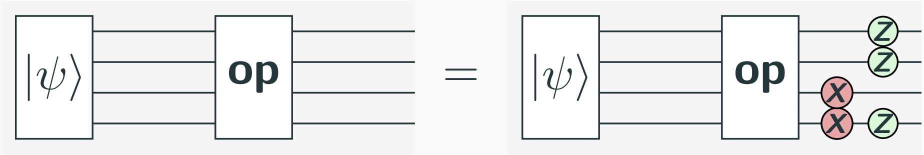
Cole Comfort and Aleks Kissinger. **"A Graphical Calculus for Lagrangian Relations".** In: (visited on 06/19/2023).

Stabiliser maps

1. 

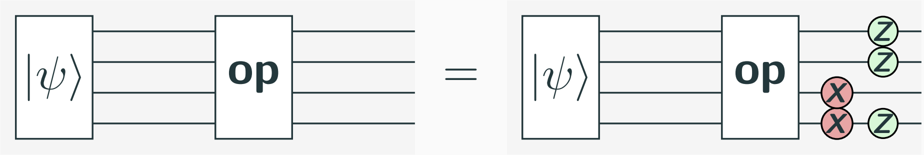
Cole Comfort and Aleks Kissinger. **"A Graphical Calculus for Lagrangian Relations".** In: (visited on 06/19/2023).

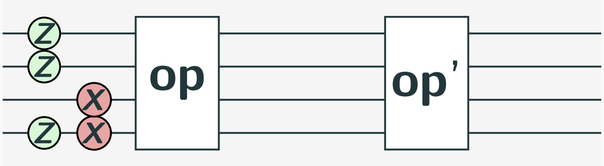
Stabiliser maps

1. 

Cole Comfort and Aleks Kissinger. **"A Graphical Calculus for Lagrangian Relations".** In: (visited on 06/19/2023).

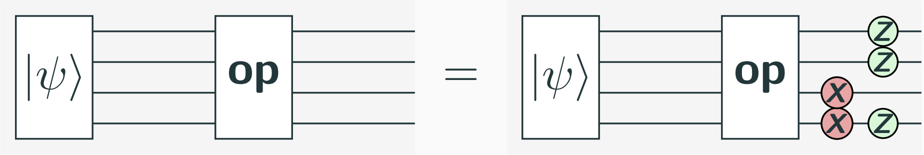
Stabiliser maps

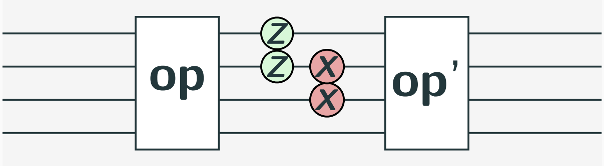
1. 

2. 

Cole Comfort and Aleks Kissinger. **“A Graphical Calculus for Lagrangian Relations”.** In: (visited on 06/19/2023).

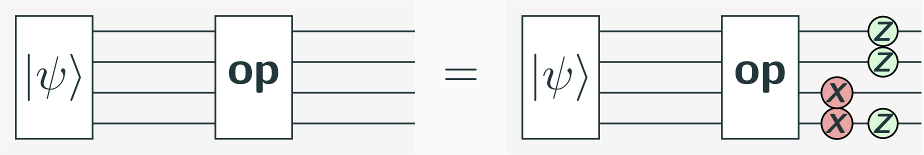
Stabiliser maps

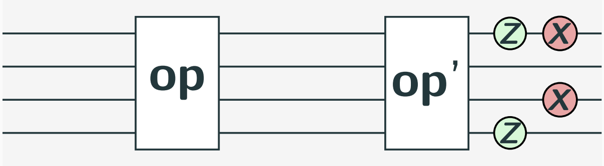
1. 

2. 

Cole Comfort and Aleks Kissinger. **"A Graphical Calculus for Lagrangian Relations"**. In: (visited on 06/19/2023).

Stabiliser maps

1. 

2. 

Cole Comfort and Aleks Kissinger. **"A Graphical Calculus for Lagrangian Relations".** In: (visited on 06/19/2023).

Semantics overview

$$\text{linear maps} \xrightarrow{\text{CPM construction}} \text{quantum channels} \xrightarrow{\text{split †-idempotents}} \text{QM with } \textit{classical types}$$

$$\mathrm{FdHilb} \longrightarrow \mathrm{MatAlg}_{CP} \longrightarrow \mathrm{FdC^*\text{-}Alg}_{CP}$$

Peter Selinger. **"Dagger Compact Closed Categories and Completely Positive Maps: (Extended Abstract)".** In: ().
Peter Selinger. **"Idempotents in Dagger Categories".** In: ().

Semantics for measurement and classical control

$$\begin{tikzcd}
\begin{array}{c}\textbf{operational}\\ = \text{Hilbert space}\end{array} & \begin{array}{c}\textbf{denotational}\\ = \text{Pauli pushing}\end{array} \\
\text{stabiliser maps} \arrow[r, leftrightarrow, "\simeq"] \arrow[d, "\text{CPM construction}"'] & \text{affine Lagrangian relations} \arrow[d, "\text{CPM construction}"] \\
\text{stabiliser channels} \arrow[r, leftrightarrow, "\simeq"] \arrow[d, "\text{split } \dagger\text{-idempotents}"'] & \text{affine coisotropic relations} \arrow[d, "\text{split } \dagger\text{-idempotents}"] \\
\begin{array}{c}\text{stabiliser channels}\\ \text{with classical/quantum types}\end{array} \arrow[r, leftrightarrow, "\simeq"] & \begin{array}{c}\text{affine coisotropic relations}\\ \text{with a classical modality}\end{array}
\end{tikzcd}$$

Summary of results

Restricting the classical control to affine operations:

Full definability

For all typed environments Γ and Δ and morphisms $f : [\![\Gamma]\!] \to [\![\Delta]\!]$ in $\mathrm{AR}_{\mathrm{qc}}$, there exists a well-formed judgement $\Gamma \vdash c \triangleright \Delta$ such that $[\![c]\!] = f$.

Full abstraction

Well-formed judgements c and d are observationally equivalent if and only if their denotations are equal $[\![c]\!] = [\![d]\!]$.

Two papers about quantum instruments

Composing Quantum Instruments

Robert I. Booth[1], Dominik Leichtle[2], Alex Rice[2], and Kim Worrall[2]

[1] *University of Oxford, United Kingdom*
[2] *University of Edinburgh, United Kingdom*

We
ural qu
integra
strume
functio
Heisenb
tensor
comple
erning
identify

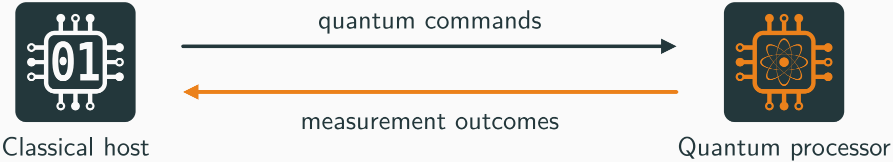
Quantur

Quantum Orchestras: a Concrete Semantics for Recursive Hybrid Programs

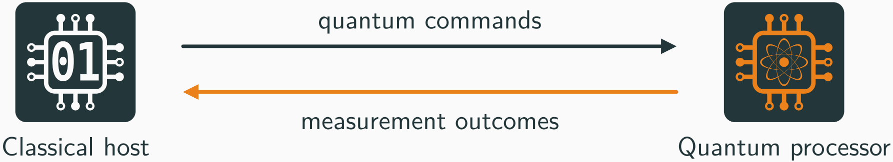
ALEX RICE, The University of Edinburgh, United Kingdom
DOMINIK LEICHTLE, The University of Edinburgh, United Kingdom
KIM WORRALL, The University of Edinburgh, United Kingdom
ROBERT I. BOOTH, University of Oxford, United Kingdom

Many production quantum programming languages represent hybrid quantum computations by extending a classical base language with a quantum effect, where qubits are addressed by reference, and quantum operations are understood to mutate some external quantum state. However, the semantics of this view of quantum computation remains underdeveloped, especially when the language allows mid-circuit measurements and non-termination.

A QEC round is a hybrid program

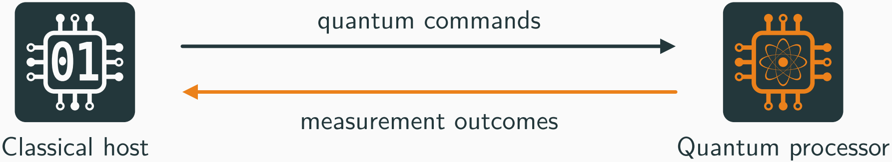


A QEC round is a hybrid program



The classical host holds data and references to qubits, while the processor retains the quantum state ρ between calls:

A QEC round is a hybrid program



The classical host holds data and references to qubits, while the processor retains the quantum state ρ between calls:

$\underline{\texttt{syndrome}} = \textbf{meas}\ \underline{\texttt{x}}\ \fatsemi$	quantum effect
$\underline{\texttt{correction}} := \mathrm{decode}(\underline{\texttt{syndrome}})\ \fatsemi$	classical computation
$\textbf{ctrl}_P\ \underline{\texttt{correction}}\ \underline{\texttt{qubits}}$	quantum effect

Quantum instruments

A quantum instrument is a (finite) family of quantum channels:

$$\mathcal{I} = (\Phi_x)_{x \in X}, \qquad \Phi_x : \mathcal{L}(H) \longrightarrow \mathcal{L}(H).$$

Each Φ_x is completely positive, and $\sum_x \Phi_x$ is trace-preserving.

Probability of outcome x

$$p_x = \mathrm{tr}[\Phi_x(\rho)]$$

State after that outcome

$$\rho_x = \frac{\Phi_x(\rho)}{p_x} \quad (p_x > 0)$$

$\mathrm{Q}(X)$ collects these interactions with classical output in X.

The instrument monad

Let $\mathcal{I} = (\Phi_x) \in \mathrm{Q}(X)$ and $K : X \longrightarrow \mathrm{Q}(Y)$ choose the next interaction.

$$\boxed{\left(\mathcal{I} \gg= K\right)_y = \sum_x K(x)_y \circ \Phi_x}$$

Bind: choose the next interaction using the observed result.

Return: return a value without changing the quantum state.

The instrument monad

Let $\mathcal{I} = (\Phi_x) \in \mathrm{Q}(X)$ and $K : X \longrightarrow \mathrm{Q}(Y)$ choose the next interaction.

$$\boxed{\left(\mathcal{I} \gg= K\right)_y = \sum_x K(x)_y \circ \Phi_x}$$

Bind: choose the next interaction using the observed result.

Return: return a value without changing the quantum state.

For syndrome measurement $\mathcal{M}$, decoder d, and corrections $\mathcal{R}_c$:

$$\mathcal{E}_d(\rho) = \sum_s \mathcal{R}_{d(s)}(\mathcal{M}_s(\rho))\,.$$

The classical decoder selects the quantum continuation.

Stabiliser instruments

Theorem 5 ([16]). *Let $\Phi : \mathfrak{T}(\mathcal{H}_A) \to \mathfrak{T}(\mathcal{H}_B)$ be a channel. The following statements are equivalent:*

1. *Φ is completely adaptive stabilizer preserving.*
2. *Choi state σ is a mixture of stabilizer states.*

The stabiliser programming language takes semantics in stabiliser instruments, i.e.

each branch Φ_x is a stabiliser circuit.

Continuous outcomes and recursion

Continuous outcomes

Meas: measurable outcome spaces

Composition by integration against an instrument.

A quantum analogue of the Giry monad.

Recursion

DCPO: ordered approximations

Quantum orchestras act continuously on continuations.

Recursive programs have least-fixed-point semantics:

$$\mathrm{lfp}(F) = \bigsqcup_{n \geqslant 0} F^n(\bot).$$

Ongoing and future work

- How do the symplectic semantics work for the qubit case?
  (ongoing work with Axel Daboust)

Ongoing and future work

- How do the symplectic semantics work for the qubit case?
  (ongoing work with Axel Daboust)

- How do we combine these two strands of work?
  e.g. to reason about QEC programs and decoders

Ongoing and future work

- How do the symplectic semantics work for the qubit case?
  (ongoing work with Axel Daboust)
- How do we combine these two strands of work?
  e.g. to reason about QEC programs and decoders
- How can we use these semantics for fault-tolerant compilation?

Ongoing and future work

- How do the symplectic semantics work for the qubit case?
  (ongoing work with Axel Daboust)
- How do we combine these two strands of work?
  e.g. to reason about QEC programs and decoders
- How can we use these semantics for fault-tolerant compilation? What is the compositional / algebraic / combinatorial structure of fault-tolerance?

CPM: Semantics of trace

CPM: Semantics of trace

$\rightsquigarrow$ the trace is just a discard!

Splitting: Semantics of measurements

Splitting: Semantics of measurements

$\rightsquigarrow$ relates the 2-dimensional space of red/green tokens to the 1-dimensional space of bit-flips