With a Few Square Roots, Quantum Computing is as Easy as Π

Jacques Carette McMaster University

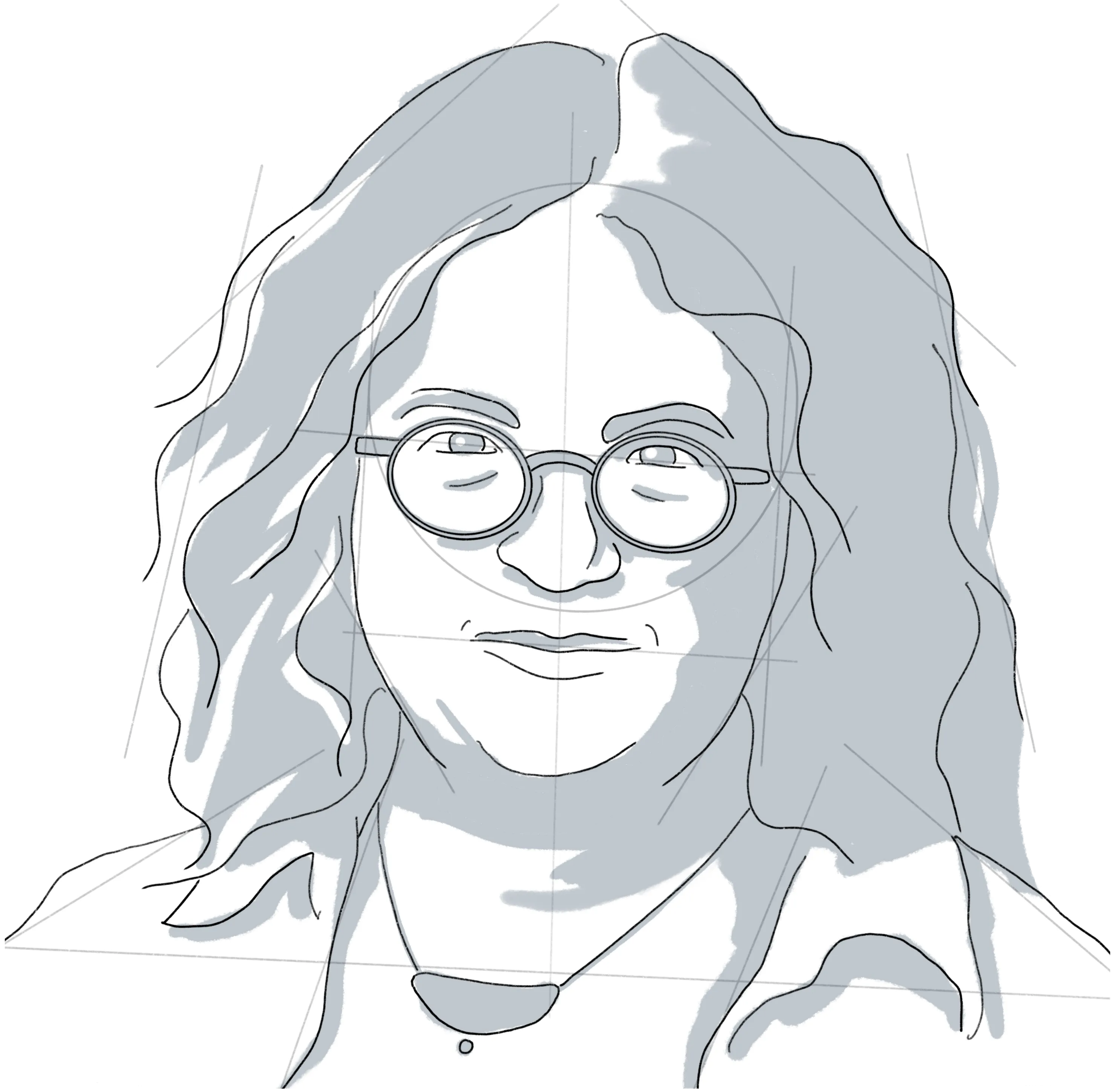
Chris Heunen University of Edinburgh

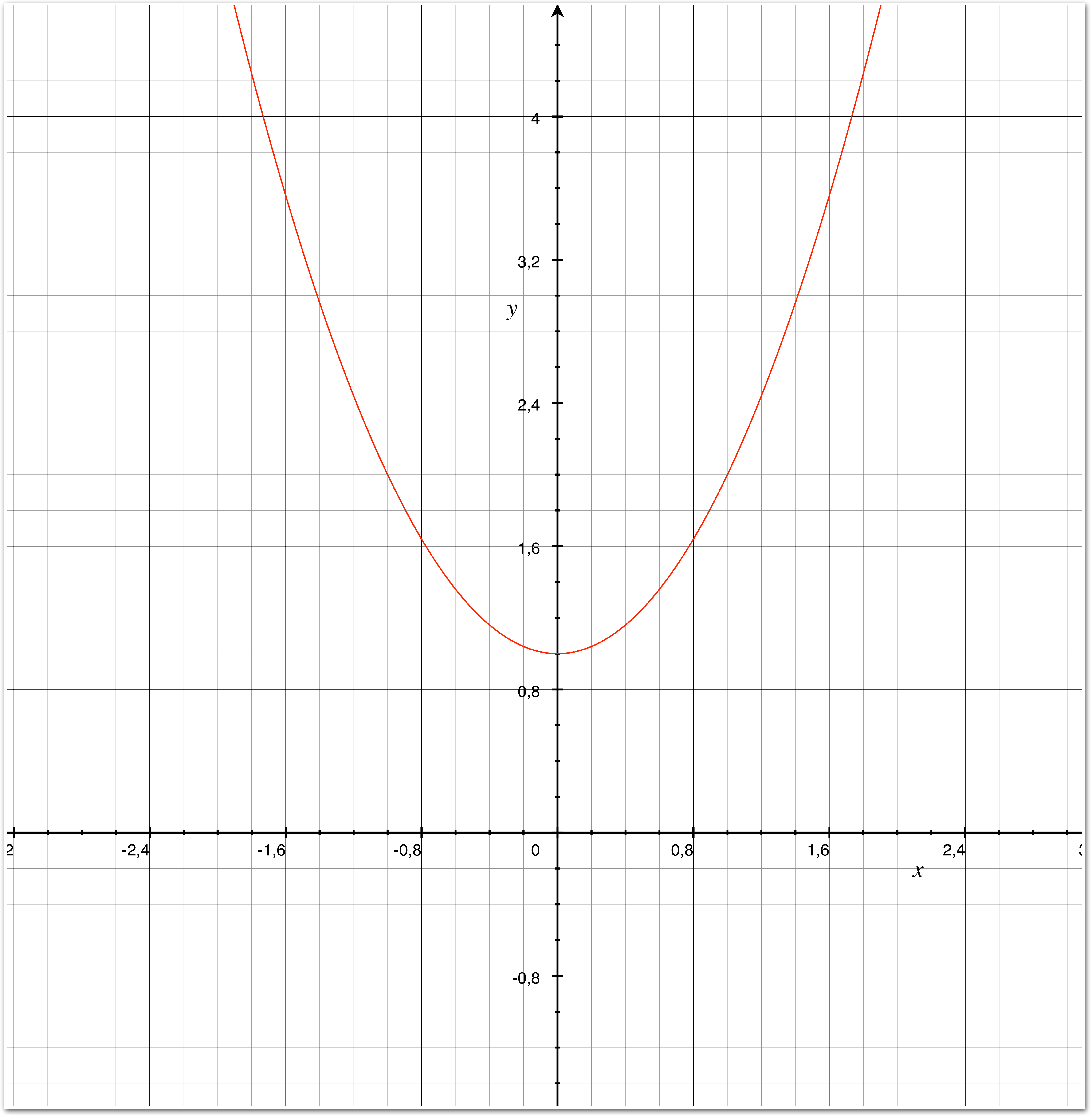
Robin Kaarsgaard University of Southern Denmark

Amr Sabry Indiana University Bloomington

Quantum Computation as a Completion

"It's really something that is special for quantum computation because it's somehow 'complete' — quantum computation is some kind of completion, mathematically, of classical computation. I think of this as maybe similar to the fact that the complex numbers are an algebraic closure of the real numbers."





Algebraic Closure

A field F is **algebraically closed** when every *(non-constant) polynomial* has a root in F.

The real numbers are famously not algebraically closed, *e.g.*, the polynomial $x^2 + 1$ has no real roots.

To get a solution to this polynomial, we need to go to the *algebraic closure* of the real numbers, *i.e.*, complex numbers.

Unitaries

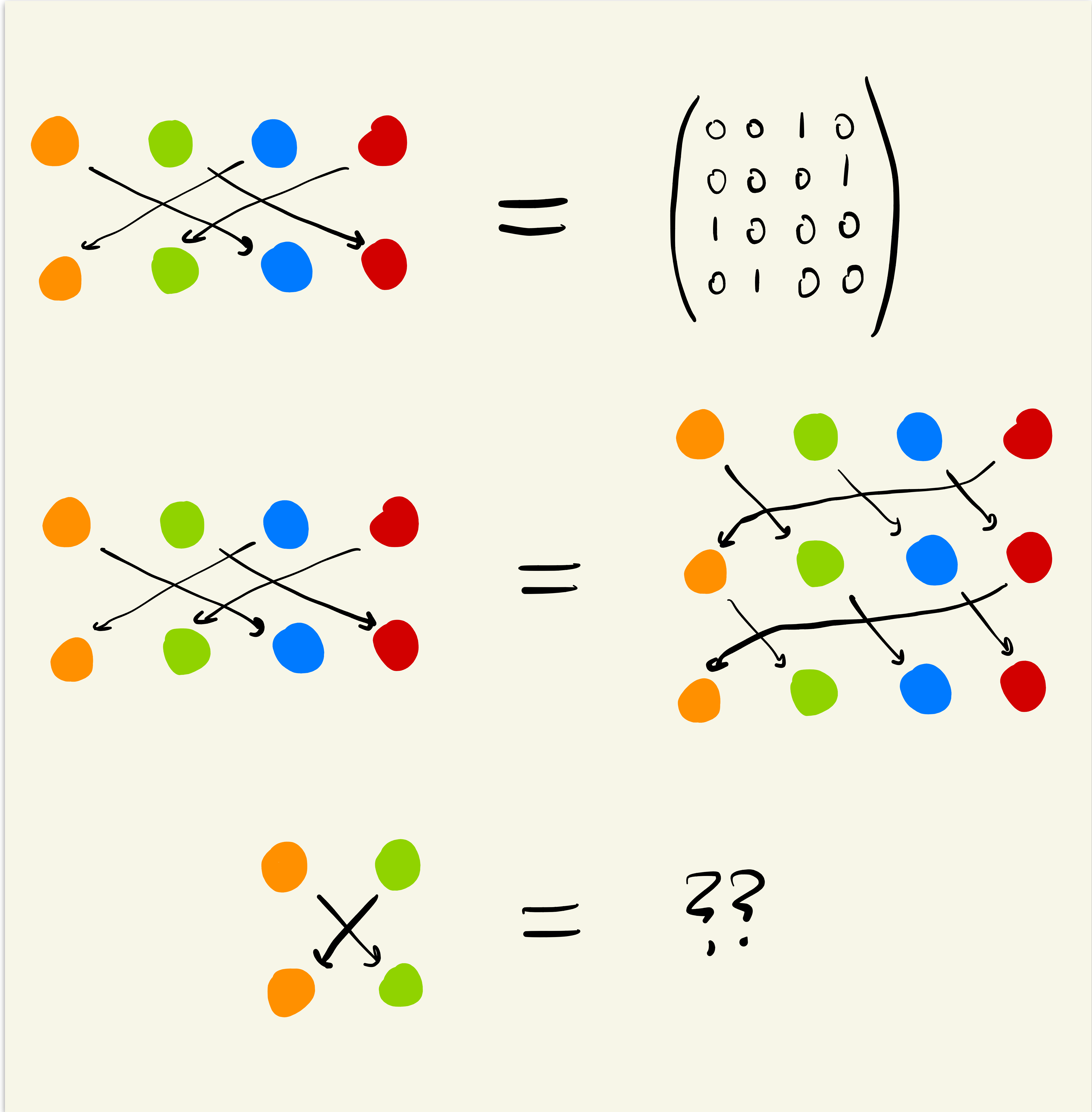
A complex $n \times n$ matrix U is **unitary** when $U^\dagger U = UU^\dagger = I$ (with $U^\dagger = \overline{U^T}$).

Unitaries give semantics to *quantum programs*.

Unitaries are a bit like numbers in that we can form polynomials over them (using matrix multiplication and entrywise sum).

Polynomials of the form $X^2 - V$ even always have unitary solutions for any unitary V – *i.e., every unitary has a (unitary) square root.*

Permutations



A permutation of n elements can be seen as an $n \times n$ unitary with *Boolean entries*.

Permutations give semantics to *reversible classical programs*.

However, not all polynomials of permutations of the form $X^2 - V$ have solutions in the permutations.

In other words, *not every permutation has a square root (in the permutations).*

A Question

Summary: Not all reversible classical programs have square roots, but all quantum programs do.

Is this a defining feature of quantum computation?

Can we recover universal quantum computation from classical reversible computation with (certain) square roots?

$$
\begin{array}{rll}
b & ::= \mathbb{0} \mid \mathbb{1} \mid b + b \mid b \times b & \text{(value types)} \\
t & ::= b \leftrightarrow b & \text{(combinator types)} \\
iso & ::= id \mid swap^{+} \mid assocr^{+} \mid assocl^{+} \mid unite^{+}l \mid uniti^{+}l \mid absorbl \mid factorzr & \text{(isomorphisms)} \\
 & \quad \mid swap^{\times} \mid assocr^{\times} \mid assocl^{\times} \mid unite^{\times}l \mid uniti^{\times}l \mid dist \mid factor & \\
c & ::= iso \mid c \fatsemi c \mid c + c \mid c \times c & \text{(combinators)}
\end{array}
$$

$$
\begin{array}{rcrcll}
id & : & b & \leftrightarrow & b & : id \\
swap^{+} & : & b_1 + b_2 & \leftrightarrow & b_2 + b_1 & : swap^{+} \\
assocr^{+} & : & (b_1 + b_2) + b_3 & \leftrightarrow & b_1 + (b_2 + b_3) & : assocl^{+} \\
unite^{+}l & : & \mathbb{0} + b & \leftrightarrow & b & : uniti^{+}l \\
swap^{\times} & : & b_1 \times b_2 & \leftrightarrow & b_2 \times b_1 & : swap^{\times} \\
assocr^{\times} & : & (b_1 \times b_2) \times b_3 & \leftrightarrow & b_1 \times (b_2 \times b_3) & : assocl^{\times} \\
unite^{\times}l & : & \mathbb{1} \times b & \leftrightarrow & b & : uniti^{\times}l \\
dist & : & (b_1 + b_2) \times b_3 & \leftrightarrow & (b_1 \times b_3) + (b_2 \times b_3) & : factor \\
absorbl & : & b \times \mathbb{0} & \leftrightarrow & \mathbb{0} & : factorzr
\end{array}
$$

$$
\frac{c_1 : b_1 \leftrightarrow b_2 \quad c_2 : b_2 \leftrightarrow b_3}{c_1 \fatsemi c_2 : b_1 \leftrightarrow b_3}
\qquad
\frac{c_1 : b_1 \leftrightarrow b_3 \quad c_2 : b_2 \leftrightarrow b_4}{c_1 + c_2 : b_1 + b_2 \leftrightarrow b_3 + b_4}
\qquad
\frac{c_1 : b_1 \leftrightarrow b_3 \quad c_2 : b_2 \leftrightarrow b_4}{c_1 \times c_2 : b_1 \times b_2 \leftrightarrow b_3 \times b_4}
$$

$$
\begin{array}{c}
\textsc{ctrl}\ c = dist \fatsemi id + (id \times c) \fatsemi factor \\
\mathsf{1} : \mathbb{1} \leftrightarrow \mathbb{1} = id \\
\textsc{x} : \mathbb{2} \leftrightarrow \mathbb{2} = swap^{+} \\
\textsc{cx} : \mathbb{2} \times \mathbb{2} \leftrightarrow \mathbb{2} \times \mathbb{2} = \textsc{ctrl}\ swap^{+} \\
\textsc{ccx} : \mathbb{2} \times \mathbb{2} \times \mathbb{2} \leftrightarrow \mathbb{2} \times \mathbb{2} \times \mathbb{2} = \textsc{ctrl}\ \textsc{cx}
\end{array}
$$

Π is for Programming with Permutations

Π is a strongly typed programming language for finite permutations.

Fact 1: Its semantics are given by *rig groupoids* and their axioms.

Fact 2: Π admits the Toffoli gate set, i.e., every finite permutation is the denotation of some Π program.

Fact 3: Π is *equationally fully abstract:* $[\![c_1]\!] = [\![c_2]\!]$ *as permutations iff* $[\![c_1]\!] = [\![c_2]\!]$ *in every rig groupoid.*

A Few Square Roots

We extend this simple language by adding two base isomorphisms

$$w : 1 \leftrightarrow 1 \qquad v : 1 + 1 \leftrightarrow 1 + 1$$

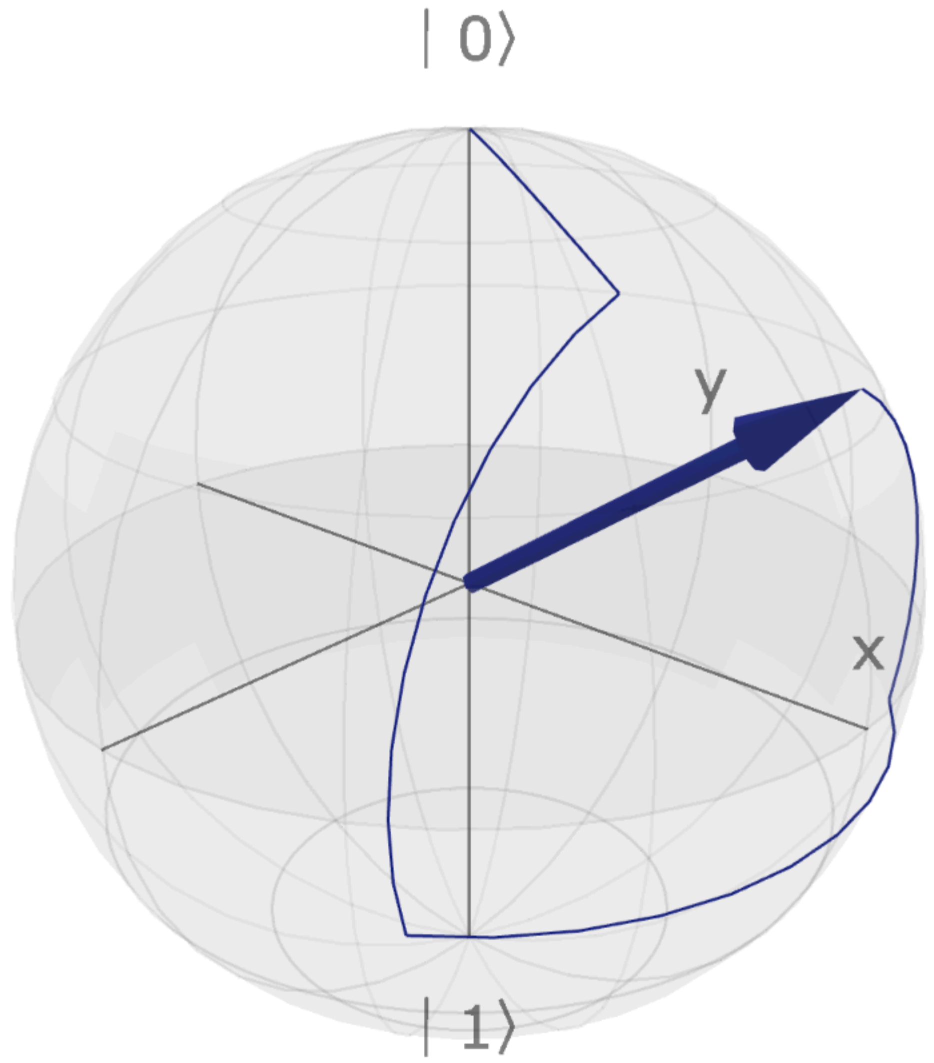
and three equations governing them

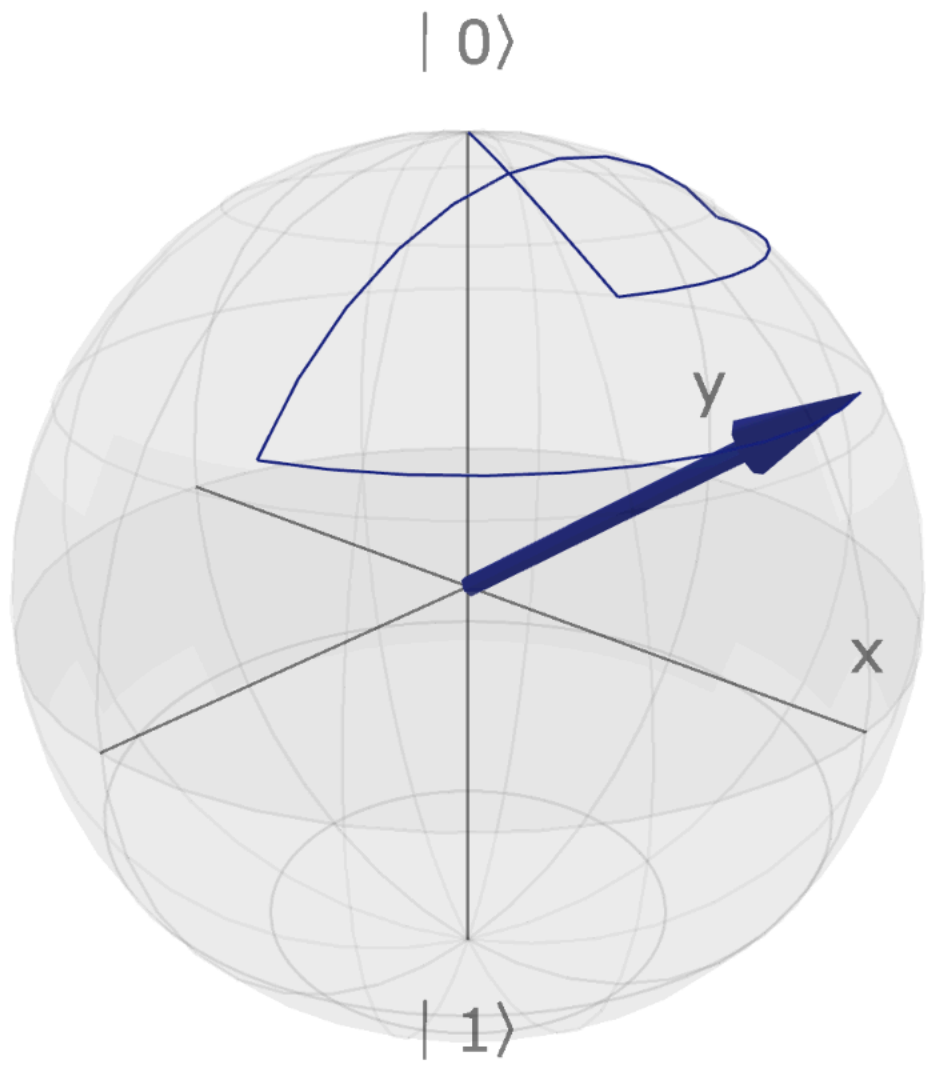
(E1) $v^2 \leftrightarrow_2 x$

(E2) $w^8 \leftrightarrow_2 \mathrm{id}_1$

(E3*) $v; s; v \leftrightarrow_2 s; v; s$ where $s = \mathrm{id} + w^2$

We call the resulting language $\sqrt{\Pi}$.





Models of $\sqrt{\Pi}$

Models of $\sqrt{\Pi}$ are *rig groupoids* $(\mathbf{C}, I, O, \otimes, \oplus)$ with distinguished maps $\omega : I \to I$ and $\mathsf{V} : I \oplus I \to I \oplus I$ satisfying the three equations.

Choosing

$$\omega = e^{2\pi i/8} \qquad \mathsf{V} = \frac{1+i}{2}\begin{pmatrix} 1 & -i \\ -i & 1 \end{pmatrix}$$

we see that **Unitary** is a model of $\sqrt{\Pi}$.

Gates and Circuits

We can represent all classical reversible gates in $\sqrt{\Pi}$, but also the *phase gates*

$$\mathsf{T} = \mathrm{id} + w \quad \mathsf{S} = \mathrm{id} + w^2 \quad \mathsf{Z} = \mathrm{id} + w^4$$

and the *Hadamard gate*

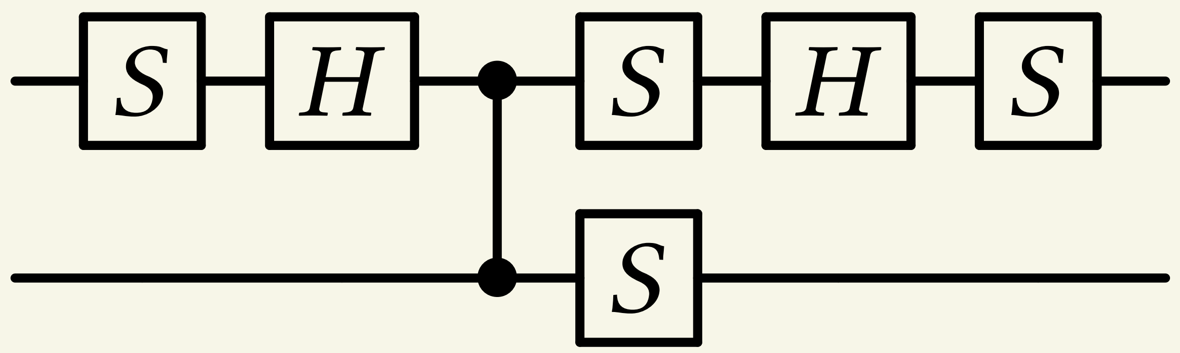
$$\mathsf{H} = w^7 \bullet (v; s; v)$$

where $s \bullet f$ denotes *abstract scalar multiplication*

$$s \bullet f = \mathrm{uniti}^{\times}\mathrm{l}; s \times f; \mathrm{unite}^{\times}\mathrm{l}$$

These coincide with usual definitions in **Unitary**.

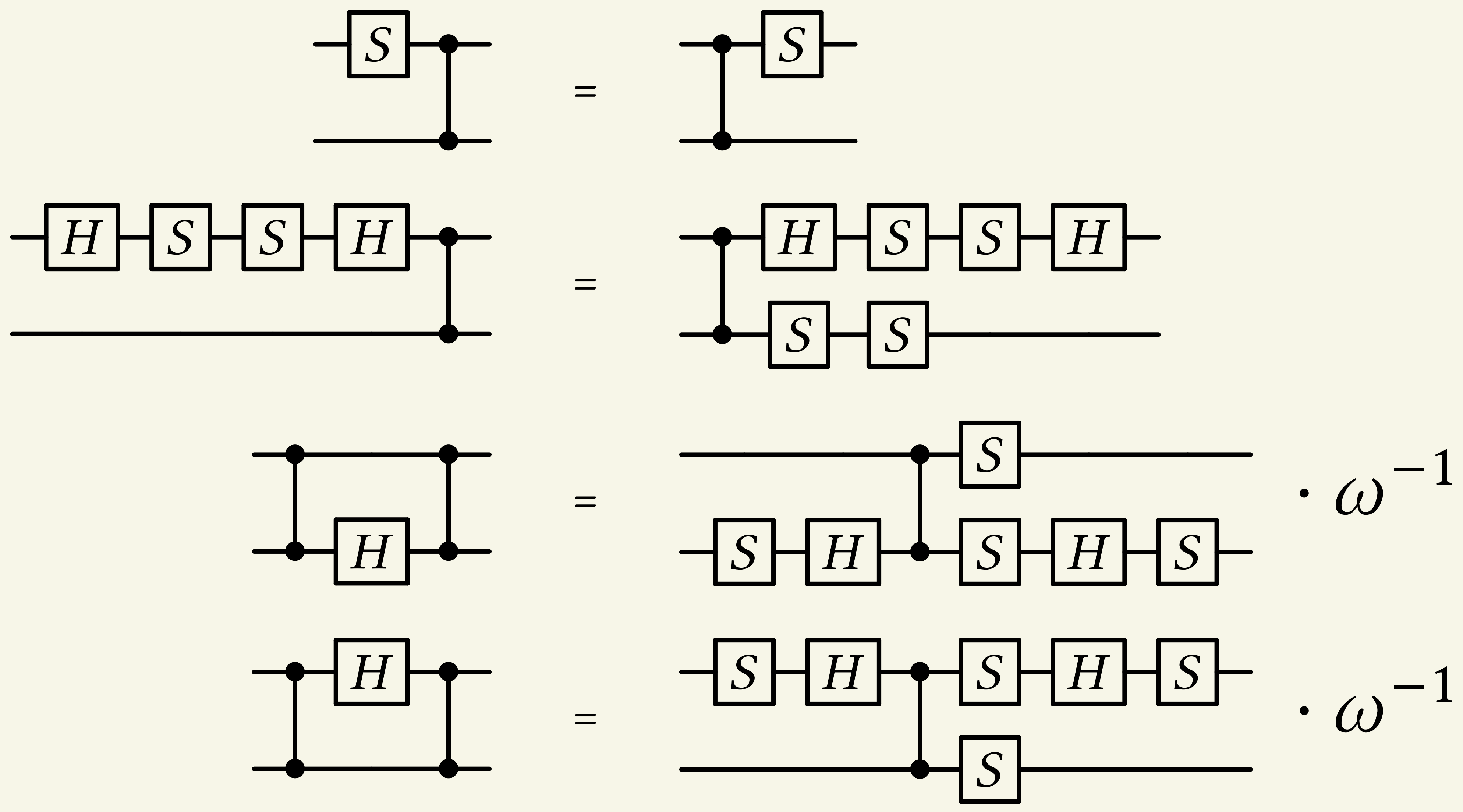
We can use this to represent circuits in various gate sets, including *Clifford+T*.



$$\Downarrow$$

$$((\mathsf{S}; \mathsf{H}) \times \mathrm{id}); \mathrm{ctrl}\ \mathsf{Z}; ((\mathsf{S}; \mathsf{H}; \mathsf{S}) \times \mathsf{S})$$

$$
\begin{aligned}
A_0 B_1 &= A_1 B_0 \\
\mathrm{H}^2 &= \mathrm{id} \\
\mathrm{SHSHSH} &= \omega \cdot \mathrm{id}
\end{aligned}
$$



Equational Theories

We can reason about these circuits using the axioms of rig groupoids and axioms (E1) — (E3).

Recent work gives sound and complete equational theories for various quantum gate sets and unitary groups.

This allows us to show *equational full abstraction* results for certain classes of terms in $\sqrt{\Pi}$.

Full Abstraction

Given any model of $\sqrt{\Pi}$, a term c has an interpretation $[\![c]\!]$. Since **Unitary** is a model $\sqrt{\Pi}$, c also has a unitary interpretation $(\!| c |\!)$.

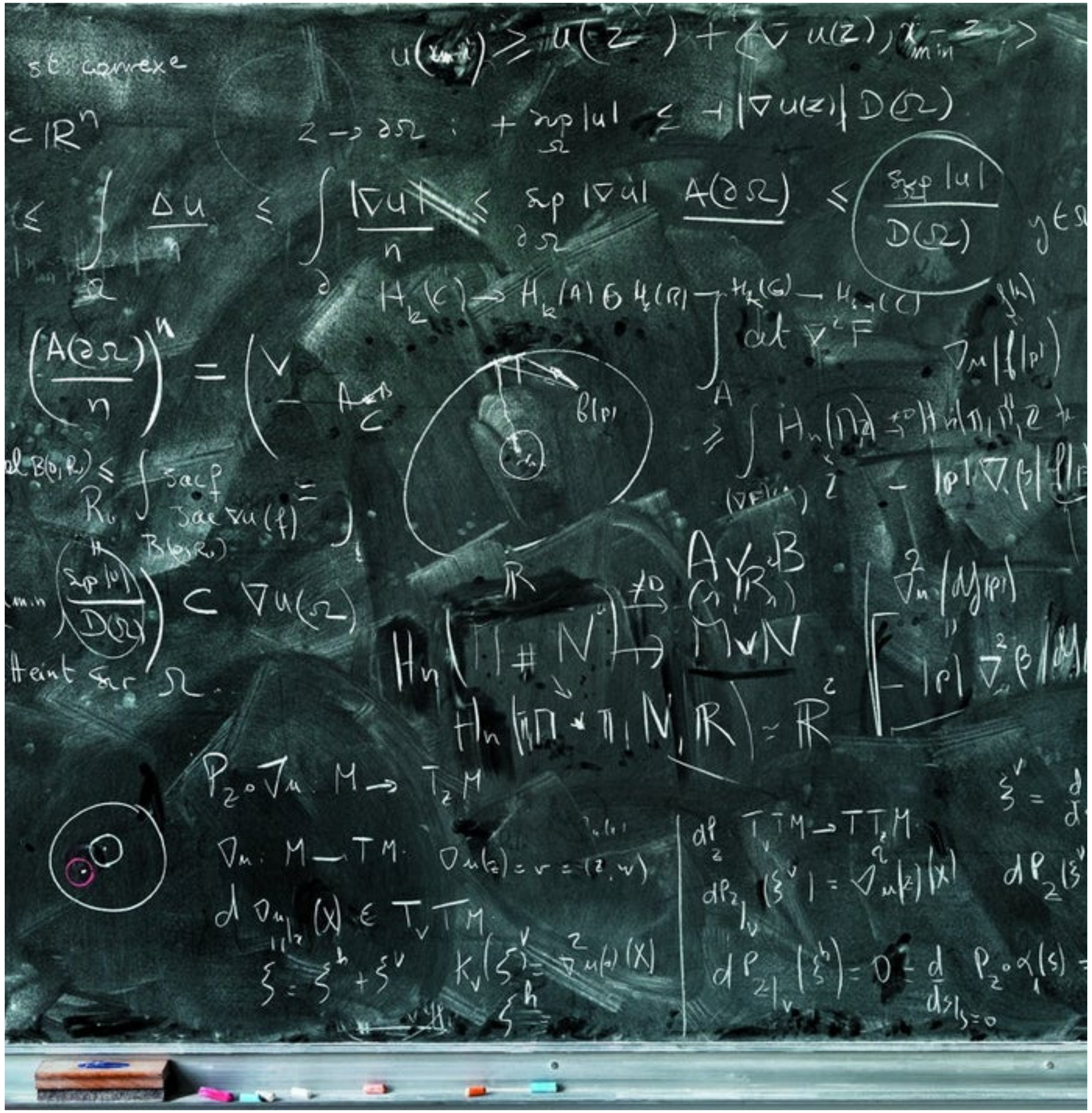
We show a number of theorems of the form

$$[\![c_1]\!] = [\![c_2]\!] \quad \text{iff} \quad (\!| c_1 |\!) = (\!| c_2 |\!)$$

for all terms c_1 and c_2 of a syntactic form, corresponding to representation of circuits formed using various gate sets.

Approach: Show that the all sound and complete equational theories in sight are implied by the rig axioms and (E1) — (E3).

$$\begin{aligned}
[\![\mathsf{S};\mathsf{S}]\!] &= [\![\mathsf{S}]\!] \circ [\![\mathsf{S}]\!] \\
&= (\mathrm{id} \oplus \omega^2) \circ (\mathrm{id} \oplus \omega^2) \\
&= (\mathrm{id} \circ \mathrm{id}) \oplus (\omega^2 \circ \omega^2) \\
&= \mathrm{id} \oplus \omega^4 \\
&= [\![\mathsf{Z}]\!]
\end{aligned}$$

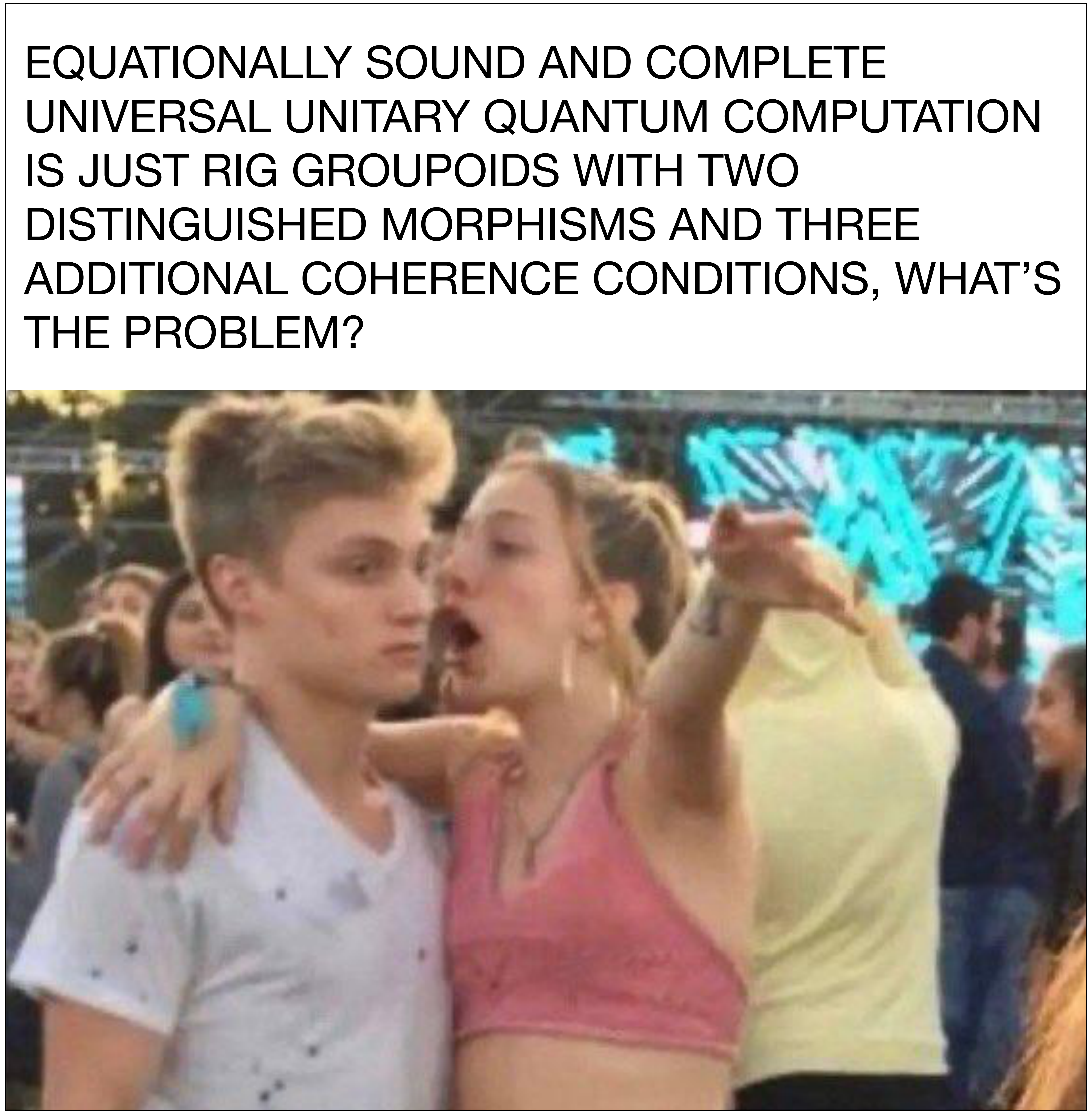


Full Abstraction

We show theorems of this form for

- Clifford circuits of arbitrary size.
- Clifford+T circuits of ≤ 2 qubits.
- Unitaries with entries in the ring $\mathbb{Z}\left[\frac{1}{2}, i\right]$.
- *Gaussian Clifford+T* (i.e., *Clifford+Toffoli*) circuits of arbitrary size.

The latter two are universal.



Closure

We have a formalisation written in Agda, including many of our lemmas and theorems, see

`https://github.com/JacquesCarette/SqrtPi/`

Central question: *Can we increase accuracy by adding more square roots and retain full abstraction?*

Roadblock: No known sound and complete equational theory for ≥2-qubit Clifford+T.

Conjecture: $\sqrt{\Pi}$ is equationally fully abstract for *all* Clifford+T circuits.

A Word From My Sponsors

We're hiring a *DIAS Tenure Track Assistant Professor* in Quantum Mathematics or Programming Technology at the University of Southern Denmark.

Topics include *quantum programming*, *quantum foundations*, and *choreographic programming*.

Deadline: March 1, 2024.

Topics: `https://www.sdu.dk/en/qmpt`

Call: `http://tinyurl.com/diasqmpt`

SDU