Higher algebra in quantum information theory

David Reutter

Department of Computer Science
University of Oxford

March 9, 2018
What is this talk about?

- **Part I**: Shaded tensor networks & biunitaries
What is this talk about?

- **Part I**: Shaded tensor networks & biunitaries
  - shaded tensor networks

- **Part II**: Untangling quantum circuits
  - a shaded tangle language for quantum circuits
  - biunitaries and error correction

Based on joint work with Jamie Vicary:
- Biunitary constructions in quantum information
- Shaded tangles for the design and verification of quantum programs

David Reutter
Higher algebra in quantum information
March 9, 2018
2 / 21
What is this talk about?

- **Part I**: Shaded tensor networks & biunitaries
  - shaded tensor networks
  - ‘biunitary’ tensors in them

Based on joint work with Jamie Vicary:

- Biunitary constructions in quantum information
- Shaded tangles for the design and verification of quantum programs

David Reutter

Higher algebra in quantum information

March 9, 2018
What is this talk about?

- **Part I**: Shaded tensor networks & biunitaries
  - shaded tensor networks
  - ‘biunitary’ tensors in them
  - composing these tensors
What is this talk about?

- **Part I:** Shaded tensor networks & biunitaries
  - shaded tensor networks
  - ‘biunitary’ tensors in them
  - composing these tensors

- **Part II:** Untangling quantum circuits

Based on joint work with Jamie Vicary:

- Biunitary constructions in quantum information
- Shaded tangles for the design and verification of quantum programs

David Reutter
Higher algebra in quantum information
March 9, 2018
What is this talk about?

- **Part I:** Shaded tensor networks & biunitaries
  - shaded tensor networks
  - ‘biunitary’ tensors in them
  - composing these tensors

- **Part II:** Untangling quantum circuits
  - a shaded tangle language for quantum circuits
What is this talk about?

- **Part I:** Shaded tensor networks & biunitaries
  - shaded tensor networks
  - ‘biunitary’ tensors in them
  - composing these tensors

- **Part II:** Untangling quantum circuits
  - a shaded tangle language for quantum circuits
  - biunitaries and error correction
What is this talk about?

- **Part I:** Shaded tensor networks & biunitaries
  - shaded tensor networks
  - ‘biunitary’ tensors in them
  - composing these tensors

- **Part II:** Untangling quantum circuits
  - a shaded tangle language for quantum circuits
  - biunitaries and error correction

Based on joint work with Jamie Vicary:

*Biunitary constructions in quantum information*

*Shaded tangles for the design and verification of quantum programs*
Part 1
Shaded tensor networks & biunitaries
Quantum structures

Let’s start with a very concrete problem.
Quantum structures

Let’s start with a very concrete problem.

Hadamard matrices $H$

$|H_{i,j}|^2 = 1 \quad H^\dagger H = n \mathbb{1}$

$(\begin{smallmatrix} 1 & 1 \\ 1 & -1 \end{smallmatrix})$
Quantum structures

Let’s start with a very concrete problem.

Hadamard matrices $H$

\[ |H_{i,j}|^2 = 1 \quad H^\dagger H = n \mathbb{1} \]

\[
\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}
\]

unitary error bases (UEB) $\{U_i\}_{1 \leq i \leq n^2}$

$U_i$ unitary \quad $\text{Tr}(U_i^\dagger U_j) = n \delta_{i,j}$

\[
(\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}, \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix})
\]
Quantum structures

Let’s start with a very concrete problem.

Hadamard matrices $H$

$|H_{i,j}|^2 = 1 \quad H^\dagger H = n\mathbb{1}$

$\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$

unitary error bases (UEB) $\{U_i\}_{1 \leq i \leq n^2}$

$U_i$ unitary \quad $\text{Tr}(U_i^\dagger U_j) = n\delta_{i,j}$

$(\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}, \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix})$

Important in quantum information ...
Quantum structures

Let’s start with a very concrete problem.

Hadamard matrices $H$

$|H_{i,j}|^2 = 1 \quad H^\dagger H = n \mathbb{1}$

$\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$

unitary error bases (UEB) \{\(U_i\)\}_{1 \leq i \leq n^2}

$U_i$ unitary \quad Tr\((U_i^\dagger U_j) = n \delta_{i,j}$

$\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \begin{pmatrix} 0 & i \\ i & 0 \end{pmatrix}, \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$

Important in quantum information ... but hard to construct.
Quantum structures

Let’s start with a very concrete problem.

Hadamard matrices $H$

$$|H_{i,j}|^2 = 1 \quad H^\dagger H = n\mathbb{1}$$

$$\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$$

unitary error bases (UEB) $\{U_i\}_{1 \leq i \leq n^2}$

$U_i$ unitary

$$\text{Tr}(U_i^\dagger U_j) = n\delta_{i,j}$$

$$\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}, \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$$

Important in quantum information ... but hard to construct.

Only a handful of known constructions, for example:

Hadamard + Hadamard + Hadamard $\leadsto$ UEB

$$\begin{pmatrix} a & b \\ c & d \end{pmatrix}$$

$$\begin{pmatrix} 1 \sqrt{n} A_{a,d} B_{b,c} C_{c,d} \end{pmatrix}$$
Quantum structures

Let’s start with a very concrete problem.

Hadamard matrices $H$

$|H_{i,j}|^2 = 1 \quad H^\dagger H = n \mathbb{1}$

$\begin{pmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \end{pmatrix}$

unitary error bases (UEB) $\{U_i\}_{1 \leq i \leq n^2}$

$U_i$ unitary $\quad \text{Tr}(U_i^\dagger U_j) = n \delta_{i,j}$

$\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}, \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$

Important in quantum information ... but hard to construct.
Only a handful of known constructions, for example:

Hadamard + Hadamard + Hadamard $\leadsto$ UEB

$(U_{ab})_{c,d} = \frac{1}{\sqrt{n}} A_{a,d} B_{b,c} C_{c,d}$

Why do they work?
Quantum structures

Let’s start with a very concrete problem.

Hadamard matrices $H$

$|H_{i,j}|^2 = 1$  \hspace{1cm} $H^\dagger H = n\mathbb{1}$

\[
\begin{pmatrix}
1 & 1 \\
1 & -1
\end{pmatrix}
\]

unitary error bases (UEB) $\{U_i\}_{1 \leq i \leq n^2}$

$U_i$ unitary \hspace{1cm} $\text{Tr}(U_i^\dagger U_j) = n\delta_{i,j}$

\[
\begin{pmatrix}
1 & 0 \\
0 & 1
\end{pmatrix}, \begin{pmatrix}
0 & -i \\
i & 0
\end{pmatrix}, \begin{pmatrix}
1 & 0 \\
0 & -1
\end{pmatrix}
\]

Important in quantum information ... but hard to construct.
Only a handful of known constructions, for example:

Hadamard + Hadamard + Hadamard $\leadsto$ UEB

\[
(U_{ab})_{c,d} = \frac{1}{\sqrt{n}} A_{a,d} B_{b,c} C_{c,d}
\]

Why do they work? Where do they come from?
Quantum structures

Let’s start with a very concrete problem.

Hadamard matrices $H$

$$|H_{i,j}|^2 = 1 \quad H^\dagger H = n\mathbb{1}$$

$$(\begin{smallmatrix} 1 & 1 \\ 1 & -1 \end{smallmatrix})$$

unitary error bases (UEB) $\{U_i\}_{1 \leq i \leq n^2}$

$U_i$ unitary

$$\text{Tr}(U_i^\dagger U_j) = n\delta_{i,j}$$

$$(\begin{smallmatrix} 1 & 0 \\ 0 & 1 \end{smallmatrix}), (\begin{smallmatrix} 0 & 1 \\ 1 & 0 \end{smallmatrix}), (\begin{smallmatrix} 0 & -i \\ i & 0 \end{smallmatrix}), (\begin{smallmatrix} 1 & 0 \\ 0 & -1 \end{smallmatrix})$$

Important in quantum information ... but hard to construct.

Only a handful of known constructions, for example:

Hadamard + Hadamard + Hadamard $\leadsto$ UEB

$$(U_{ab})_{c,d} = \frac{1}{\sqrt{n}} A_{a,d} B_{b,c} C_{c,d}$$

Why do they work? Where do they come from? How can we find them?
Quantum structures

Let’s start with a very concrete problem.

Hadamard matrices $H$

\[ \begin{align*}
|H_{i,j}|^2 &= 1 \\
H^\dagger H &= n1
\end{align*} \]

unitary error bases (UEB) $\{U_i\}_{1 \leq i \leq n^2}$

\[ U_i \text{ unitary} \quad \text{Tr}(U_i^\dagger U_j) = n\delta_{i,j} \]

Important in quantum computing, but hard to construct.

Only a handful of known constructions, for example:

Hadamard + Hadamard + Hadamard $\Rightarrow$ UEB

\[ \begin{align*}
(U_{ab})_{c,d} &= \frac{1}{\sqrt{n}} A_{a,d} B_{b,c} C_{c,d}
\end{align*} \]

An algebraic problem?

Why do they work? Where do they come from? How can we find them?
A higher algebraic problem!
What is higher algebra?

- Ordinary algebra lets us compose along a line:
  \[ xy^2 zyx^3 \]
What is higher algebra?

- Ordinary algebra lets us compose along a line:
  \[ xy^2 zyx^3 \]

- Higher algebra lets us compose in higher dimensions:
Planar algebra = 2-category theory

The language describing algebra in the plane is 2-category theory:

$$
\begin{align*}
A & \quad \text{objects} \\
A \xrightarrow{f} B & \quad \text{1-morphism} \\
A \xrightarrow[\eta]{g} B & \quad \text{2-morphism}
\end{align*}
$$
Planar algebra = 2-category theory

The language describing algebra in the plane is 2-category theory:

\[
\begin{array}{ccc}
A & \rightarrow & B \\
\eta & \circlearrowright & \epsilon
\end{array}
\]

objects \hspace{1cm} 1-morphism \hspace{1cm} 2-morphism

We can compose 2-morphisms like this:

\[
\begin{array}{ccc}
A & \rightarrow & B \\
\eta & \circlearrowright & \eta
\end{array}
\hspace{2cm}
\begin{array}{ccc}
A & \circlearrowright & B \\
\eta & \circlearrowright & \epsilon
\end{array}
\]

vertical composition \hspace{2cm} horizontal composition

These are pasting diagrams.
Planar algebra = 2-category theory

The language describing algebra in the plane is 2-category theory:

\[ \begin{array}{c}
\text{objects} \\
A \\
B
\end{array} \quad \begin{array}{c}
\text{1-morphism} \\
A \overset{f}{\rightarrow} B
\end{array} \quad \begin{array}{c}
\text{2-morphism} \\
A \overset{g}{\Rightarrow} B
\end{array} \]

We can compose 2-morphisms like this:

\[ \begin{array}{c}
\text{vertical composition} \\
A \overset{\eta}{\rightarrow} B
\end{array} \quad \begin{array}{c}
\text{horizontal composition} \\
A \overset{\eta}{\Rightarrow} B \overset{\epsilon}{\Rightarrow} C
\end{array} \]

These are pasting diagrams.
The dual diagrams are the graphical calculus.
Monoidal dagger pivotal 2-categories

We use *monoidal dagger pivotal 2-categories*:
Monoidal dagger pivotal 2-categories

We use *monoidal dagger pivotal* 2-categories:

- Dagger pivotal 2-categories have a very flexible graphical calculus.

![Diagram](image)
Monoidal dagger pivotal 2-categories

We use *monoidal dagger pivotal* 2-categories:

- Dagger pivotal 2-categories have a very flexible graphical calculus.
- In a monoidal 2-category, we can layer surfaces on top of each other.
We use *monoidal dagger pivotal 2-categories*:
- Dagger pivotal 2-categories have a very flexible graphical calculus.
- In a monoidal 2-category, we can layer surfaces on top of each other.

⇒ surfaces, wires and vertices in three-dimensional space
A model for quantum computation: \( \mathbf{2Hilb} \)

We work in the 2-category \( \mathbf{2Hilb} \), a categorification of \( \mathbf{Hilb} \).
A model for quantum computation: $\mathbf{2Hilb}$

We work in the 2-category $\mathbf{2Hilb}$, a categorification of $\mathbf{Hilb}$.
- Objects are natural numbers $n, m, ...$
A model for quantum computation: \( \mathcal{2}\text{Hilb} \)

We work in the 2-category \( \mathcal{2}\text{Hilb} \), a categorification of \( \text{Hilb} \).

- Objects are natural numbers \( n, m, \ldots \)
- 1-morphisms \( n \xrightarrow{H} m \) are matrices of Hilbert spaces

\[
\begin{pmatrix}
H_{11} & \cdots & H_{1n} \\
\vdots & \ddots & \vdots \\
H_{m1} & \cdots & H_{mn}
\end{pmatrix}
\]

This well-studied structure plays a key role in higher representation theory.
We work in the 2-category $2\text{Hilb}$, a categorification of $\text{Hilb}$.  

- Objects are natural numbers $n, m, ...$
- 1-morphisms $n \xrightarrow{H} m$ are matrices of Hilbert spaces
- 2-morphisms $H \xrightarrow{\phi} H'$ are matrices of linear maps

$$
\begin{pmatrix}
H_{11} & \cdots & H_{1n} \\
\vdots & \ddots & \vdots \\
H_{m1} & \cdots & H_{mn}
\end{pmatrix}
\quad
\begin{pmatrix}
\begin{pmatrix}
H_{11} & \cdots & H_{1n} \\
\vdots & \ddots & \vdots \\
H_{m1} & \cdots & H_{mn}
\end{pmatrix}
\xrightarrow{11}
\begin{pmatrix}
H'_{11} & \cdots & H'_{1n} \\
\vdots & \ddots & \vdots \\
H'_{m1} & \cdots & H'_{mn}
\end{pmatrix}
\end{pmatrix}
$$

This well-studied structure plays a key role in higher representation theory.
A model for quantum computation: 2Hilb

We work in the 2-category $2\text{Hilb}$, a categorification of $\text{Hilb}$.

- Objects are natural numbers $n, m, ...$
- 1-morphisms $n \xrightarrow{H} m$ are matrices of Hilbert spaces
- 2-morphisms $H \Rightarrow H'$ are matrices of linear maps

$$
\begin{pmatrix}
H_{11} & \cdots & H_{1n} \\
\vdots & \ddots & \vdots \\
H_{m1} & \cdots & H_{mn}
\end{pmatrix}
\Rightarrow
\begin{pmatrix}
H_{11} \xrightarrow{\phi_{11}} H'_{11} & \cdots & H_{1n} \xrightarrow{\phi_{1n}} H'_{1n} \\
\vdots & \ddots & \vdots \\
H_{m1} \xrightarrow{\phi_{m1}} H'_{m1} & \cdots & H_{mn} \xrightarrow{\phi_{mn}} H'_{mn}
\end{pmatrix}
$$

This well-studied structure plays a key role in higher representation theory.
A direct perspective: tensor networks

vector space \( V \)

linear map \( F : V \to W \)

A (composed) linear map
\[ E \otimes F \to A \]
A direct perspective: shaded tensor networks

A family of linear maps, indexed by $i$ and $j$

$E \otimes F \to A$

$B \to L$

$C \to M$

$D \to N$
A direct perspective: shaded tensor networks

indexing set $i \in S$

family of vector spaces $V_i, j$

family of linear maps $F_{i, j}: V_i, j \rightarrow W_i, j$

$A \otimes B \rightarrow C$
A direct perspective: shaded tensor networks

indexing set $i \in S$

family of vector spaces $V_{i,j}$

family of linear maps $F_{i,j}: V_{i,j} \rightarrow W_{i,j}$
A direct perspective: shaded tensor networks

indexing set
\[ i \in S \]

family of vector spaces \( V_{i,j} \)

family of linear maps \( F_{i,j} : V_{i,j} \to W_{i,j} \)

A family of linear maps, indexed by \( i \) and \( j \)

\[ A_i \otimes F_j \to A_i \]

\[ B_i \otimes D_j \to C \]

\[ M_{i,j} \otimes L_i \to C \]

\[ E_{i,j} \otimes F_j \to F_j \]
A direct perspective: shaded tensor networks

- Indexing set: \( i \in S \)
- Family of vector spaces: \( V_{i,j} \)
- Family of linear maps: \( F_{i,j} : V_{i,j} \to W_{i,j} \)

A family of linear maps, indexed by \( i \) and \( j \):

\[
E_{i,j} \otimes F_j \to A_i
\]
A direct perspective: shaded tensor networks

indexing set $i \in S$

family of vector spaces $V_{i,j}$

family of linear maps $F_{i,j} : V_{i,j} \to W_{i,j}$

A family of linear maps, indexed by $i$

$$E_{i,j} \otimes F_j \to A_i$$
Biunitarity

A biunitary is a 2-morphism that is
A biunitary is a 2-morphism that is

- *(vertically*) unitary:

\[ U^\dagger U = U U^\dagger = \lambda U^\dagger U = \lambda U \]

These look just like the second Reidemeister move.
A biunitary is a 2-morphism that is

- **(vertically) unitary:**

\[
\begin{align*}
U^{\dagger} U & = U U^{\dagger} \\
\end{align*}
\]

- **horizontally unitary:**

\[
\begin{align*}
U U^{\dagger} & = \lambda \\
U^{\dagger} U & = \lambda \\
\end{align*}
\]
A biunitary is a 2-morphism that is

- (vertically) unitary:

- horizontally unitary:

These look just like the second Reidemeister move.
Quantum structures are biunitaries in $\mathbb{2Hilb}$

**Result 1:** Hadamards and UEBs are biunitaries of the following type:

![Hadamard](image1.png)  
**Hadamard**

![UEB](image2.png)  
**UEB**
Quantum structures are biunitaries in $\mathcal{2Hilb}$

**Result 1:** Hadamards and UEBs are biunitaries of the following type:

![Hadamard and UEB diagrams](image)

**Result 2:** We can compose biunitaries diagonally:
Quantum structures are biunitaries in $2\text{Hilb}$

**Result 1:** Hadamards and UEBs are biunitaries of the following type:

\[ H \quad \text{Hadamard} \quad \text{UEB} \quad U \]

**Result 2:** We can compose biunitaries diagonally:
Composing quantum structures

\[(H \otimes U)_{a,b}^{c,d} = \frac{1}{\sqrt{n}} \cdot (H_{a,b} \otimes U_{c,d})_{a,b}^{c,d} \]

\[
\begin{array}{c}
\text{Had} \\
\text{UEB}
\end{array}
\]

\[
\begin{array}{c}
A \\
B \\
C \\
A
\end{array}
\]
Composing quantum structures

$$\text{Had}$$

$$\text{UB}$$
Composing quantum structures

\[ \text{Had} + \text{Had} \]

Diagram showing two quantum structures labeled ‘Had’ and ‘UEB’.
Composing quantum structures

\[ \text{Had} + \text{Had} + \text{Had} \]

\[
(A \otimes B \otimes C) = \frac{1}{\sqrt{n}} (a \otimes b \otimes c)
\]
Composing quantum structures

Had + Had + Had \rightsquigarrow UEB
Composing quantum structures

\[
\text{Had} + \text{Had} + \text{Had} \sim \text{UEB}
\]

\[
(U_{ab})_{c,d} = \frac{1}{\sqrt{n}} A_{a,d} B_{b,c} C_{c,d}
\]
Composing quantum structures

Had + Had + Had $\rightsquigarrow$ UEB

$$(U_{ab})_{c,d} = \frac{1}{\sqrt{n}} A_{a,d} B_{b,c} C_{c,d}$$
Composing biunitaries

\[ U_{abc,de,fg} = H_{a,eg}^b, c P_{e,b,f}^c,g Q_{c,g,d} \]

\[ U_{abc,def,gh} = \sum_r V_{a,rf,g}^b,c Q_{b,r,d}^c W_{rc,e,h} \]

\[ U_{abc,de,fg} = \sum_r H_{a,r}^b,c P_{c,r,d} Q_{r,b,f} V_{r,e,g} \]

\[ U_{abcd,ef,gh} = \frac{1}{n} \sum_{r,s} A_{f,h}^r B_{s,f}^r C_{r,h} D_{s,r} H_{a,s}^d K_{b,r}^c Q_{d,s,e} P_{r,c,g} \]
Composing biunitaries

\[ U_{\alpha \beta \gamma, \delta \epsilon \zeta} = H_{a, \epsilon \gamma}^{b, c} P_{e, b, f}^{c, g} Q_{c, r, d} \]

\[ U_{\alpha \beta \gamma, \delta \epsilon \zeta} = \sum_r V_{a, r, f}^{b, c} Q_{b, r, d}^{c, e} W_{r, c, e}^{d, h} \]

\[ U_{\alpha \beta \gamma, \delta \epsilon \zeta} = \sum_r H_{a, r}^{b, c} P_{c, r, d}^{b, f} Q_{r, b, f}^{c, g} \]

\[ U_{\alpha \beta \gamma, \delta \epsilon \zeta} = \frac{1}{n} \sum_{r, s} A_{f, h}^{a, s} B_{s, f}^{C_r, h} D_{s, r}^{D_s, r} H_{a, s}^{d, s} K_{b, r}^{c, e} Q_{d, s, e}^{P_{r, c}} \]
Taking a step back

- Tensor networks:
  
  *see structural properties hidden in conventional matrix notation*
Taking a step back

- Tensor networks:  
  see structural properties hidden in conventional matrix notation
- Shaded tensor networks:  
  see structural properties hidden in tensor network notation
Taking a step back

- Tensor networks:
  see structural properties hidden in conventional matrix notation
- Shaded tensor networks:
  see structural properties hidden in tensor network notation

⇒ harness combinatorial richness of planar geometry
Taking a step back

- Tensor networks:
  see structural properties hidden in conventional matrix notation
- Shaded tensor networks:
  see structural properties hidden in tensor network notation
  ⇒ harness combinatorial richness of planar geometry

But now enough of linear algebra and let’s have some fun!
Taking a step back

- Tensor networks:
  see structural properties hidden in conventional matrix notation
- Shaded tensor networks:
  see structural properties hidden in tensor network notation

⇒ harness combinatorial richness of planar geometry

But now enough of linear algebra and let’s have some fun!

Recall:

Hadamard matrix
Taking a step back

- Tensor networks: see structural properties hidden in conventional matrix notation
- Shaded tensor networks: see structural properties hidden in tensor network notation
  \[ \Rightarrow \text{harness combinatorial richness of planar geometry} \]

But now enough of linear algebra and let’s have some fun!

Recall:

Hadamard matrix \( \leadsto \)
Taking a step back

- Tensor networks:
  see structural properties hidden in conventional matrix notation
- Shaded tensor networks:
  see structural properties hidden in tensor network notation
⇒ harness combinatorial richness of planar geometry

But now enough of linear algebra and let’s have some fun!

Recall:

Hadamard matrix $\sim \rightarrow H \sim \rightarrow$
Part 2
Untangling quantum circuits
Basic states and gates

$|+\rangle = |0\rangle + |1\rangle$  
$|\text{Bell}\rangle = |00\rangle + |11\rangle$  
$|\text{GHZ}\rangle = |000\rangle + |111\rangle$
Basic states and gates

$$|+\rangle = |0\rangle + |1\rangle \quad |\text{Bell}\rangle = |00\rangle + |11\rangle \quad |\text{GHZ}\rangle = |000\rangle + |111\rangle$$

$$|i\rangle \mapsto \sum_{j} H_{ij} |j\rangle \quad |i\rangle \otimes |j\rangle \mapsto H_{ij} |i\rangle \otimes |j\rangle$$
Basic states and gates

\[ |+\rangle = |0\rangle + |1\rangle \quad |\text{Bell}\rangle = |00\rangle + |11\rangle \quad |\text{GHZ}\rangle = |000\rangle + |111\rangle \]

Hadamard gate

CZ gate
Creating GHZ states

How to create a GHZ state from $|+\rangle$ states?
How to create a GHZ state from $|+\rangle$ states?
How to create a GHZ state from $|+\rangle$ states?
Creating GHZ states

How to create a GHZ state from $|+\rangle$ states?
Creating GHZ states

How to create a GHZ state from $|+\rangle$ states?
Creating GHZ states

How to create a GHZ state from $|+_\rangle$ states?

\[ \text{Diagram of GHZ state creation} \]
Creating GHZ states

How to create a GHZ state from $|+\rangle$ states?

\[ \begin{array}{c}
\text{CZ} \\
\text{H} \\
\text{CZ}
\end{array} \]
Creating GHZ states

How to create a GHZ state from $|+\rangle$ states?

$$|\text{GHZ}\rangle = |+\rangle |+\rangle |+\rangle$$
Quantum error correction

A $k$-local quantum code is an isometry $H \overset{\text{enc}}{\rightarrow} H^{\otimes n}$, s.t.

$$H \overset{\text{enc}}{\rightarrow} H^{\otimes n} \overset{E}{\rightarrow} H^{\otimes n} \overset{\text{enc}^\dagger}{\rightarrow} H$$

is proportional to the identity for every $k$-local error $E : H^{\otimes n} \rightarrow H^{\otimes n}$. 
Quantum error correction

A $k$-local quantum code is an isometry $H \xrightarrow{\text{enc}} H^\otimes n$, s.t.

$$H \xrightarrow{\text{enc}} H^\otimes n \xrightarrow{E} H^\otimes n \xrightarrow{\text{enc}^\dagger} H$$

is proportional to the identity for every $k$-local error $E : H^\otimes n \to H^\otimes n$. 

phase error
Quantum error correction

A $k$-local quantum code is an isometry $H \xrightarrow{\text{enc}} H^\otimes n$, s.t.

$$H \xrightarrow{\text{enc}} H^\otimes n \xrightarrow{E} H^\otimes n \xrightarrow{\text{enc}^\dagger} H$$

is proportional to the identity for every $k$-local error $E : H^\otimes n \rightarrow H^\otimes n$.

phase error

full error
The phase code

The following is a 2–local phase error code $H \rightarrow H \otimes^3$:
The phase code

The following is a $2$–local phase error code $H \rightarrow H^\otimes 3$: 

![Diagram]

New construction of a phase code from unitary error bases.
The phase code

The following is a 2–local phase error code $H \to H^\otimes 3$:
The phase code

The following is a 2–local phase error code $H \rightarrow H^\otimes 3$:
The phase code

The following is a 2–local phase error code $H \to H^\otimes 3$.

\[
\begin{array}{c}
\begin{array}{c}
\text{RII}
\end{array}
\end{array}
\sim
\begin{array}{c}
\begin{array}{c}
\text{RII}
\end{array}
\end{array}
\sim
\begin{array}{c}
\begin{array}{c}
\text{RII}
\end{array}
\end{array}
\end{array}
\]
The phase code

The following is a 2–local phase error code $H \rightarrow H \otimes^3$:
The phase code

The following is a 2–local phase error code $H \to H^\otimes 3$:
The phase code

The following is a 2–local phase error code $H \rightarrow H \otimes^3$: 

\begin{align*}
\begin{array}{c}
\begin{array}{c}
\text{RII} \\
\sim \\
\text{RII} \\
\sim
\end{array}
\end{array}
\end{align*}

\begin{align*}
\begin{array}{c}
\begin{array}{c}
\text{RII} \\
\sim \\
\text{RII} \\
\sim
\end{array}
\end{array}
\end{align*}
The phase code

The following is a 2–local phase error code $H \rightarrow H^\otimes 3$:

New construction of a phase code from unitary error bases.
Future work: The 5-qubit code

A 2–local full error correcting code $H \rightarrow H^{\otimes 5}$:
Future work: The 5-qubit code
Future work: The 5-qubit code
Future work: The 5-qubit code

Caveat: We cannot yet handle two non-adjacent errors.
Future work: The 5-qubit code
Future work: The 5-qubit code

Caveat: We cannot yet handle two non-adjacent errors.
Future work: The 5-qubit code

Caveat: We cannot yet handle two non-adjacent errors.

Thanks for listening!