Picturing Quantum Processes

Aleks Kissinger

QTFT, Växjö 2015

June 10, 2015
Quantum Picturalism: what it **is**, what it **isn’t**

- ‘QPism’ 😊 is a *methodology* for **expressing**, **teaching**, and **reasoning** about quantum processes
Quantum Picturalism: what it is, what it isn’t

- ‘QPism’ is a *methodology* for **expressing**, **teaching**, and **reasoning** about quantum processes
- *Diagrams* live at the centre, thus **composition** and **interaction**
Quantum Picturalism: what it is, what it isn’t

• ‘QPism’ ☺ is a methodology for expressing, teaching, and reasoning about quantum processes
• Diagrams live at the centre, thus composition and interaction
• QP is not a reconstruction, but some ideas from operational reconstructions play a major role, e.g.

\[
\begin{align*}
\begin{array}{c}
\rho\langle\rho'\mid\mu
\end{array}
\begin{array}{c}
\Phi
\end{array}
\begin{array}{c}
\rho
\end{array}
\end{align*}
\]
local/process tomography

\[
\begin{array}{c}
\Phi
\end{array}
\begin{array}{c}
=\hat{f}
\end{array}
\]
purification
Quantum Picturalism: what it is, what it isn’t

- ‘QPism’ 😊 is a *methodology* for **expressing**, **teaching**, and **reasoning** about quantum processes
- *Diagrams* live at the centre, thus **composition** and **interaction**
- QP is not a **reconstruction**, but some ideas from operational reconstructions play a major role, e.g.

\[
\begin{align*}
\Phi &= \hat{\mathcal{F}} \\
\end{align*}
\]

local/process tomography

purification

- …and relationship between **operational setups** and **theoretical models**: 

\[
\begin{align*}
\begin{bmatrix}
\end{align*}
\]

\[
\begin{align*}
\end{align*}
\]

Picturing Quantum Processes
A first course in quantum theory and diagrammatic reasoning

Bob Coecke & Aleks Kissinger
CUP 2015
Outline

Picturing Quantum Processes
chapters 4-9 (roughly)
Outline

Picturing Quantum Processes
chapters 4-9 (roughly)

1. Process theory of **linear maps**
Outline

Picturing Quantum Processes
chapters 4-9 (roughly)

1. Process theory of **linear maps**
2. **quantum maps** via ‘doubling’ construction
Outline

Picturing Quantum Processes
chapters 4-9 (roughly)

1. Process theory of **linear maps**
2. **quantum maps** via ‘doubling’ construction
3. Consequences: purification, causality, no-signalling, no-broadcasting
Outline

Picturing Quantum Processes
chapters 4-9 (roughly)

1. Process theory of **linear maps**
2. **quantum maps** via ‘doubling’ construction
3. Consequences: purification, causality, no-signalling, no-broadcasting
4. Classical/quantum interaction
Outline

Picturing Quantum Processes
chapters 4-9 (roughly)

1. Process theory of **linear maps**
2. **quantum maps** via ‘doubling’ construction
3. Consequences: purification, causality, no-signalling, no-broadcasting
4. Classical/quantum interaction
5. Complementarity
Recap

- Wires represent *systems*, boxes represent *processes*
Recap

- Wires represent *systems*, boxes represent *processes*

- The world is organised into *process theories*, collections of processes that make sense to combine into *diagrams*
Recap

- Certain processes play a special role:

\[
\begin{align*}
\text{states: } & \psi \\
\text{effects: } & \phi \\
\text{numbers: } & \lambda
\end{align*}
\]
Recap

- Certain processes play a special role:

\[
\text{states: } \psi \quad \text{effects: } \phi \quad \text{numbers: } \lambda
\]

- State + effect = number, interpreted as:

\[
\text{test}\left\{\begin{array}{c}
\phi \\
\psi
\end{array}\right\} \quad \text{probability}
\]

this is called the *Born rule*.
In the process theory of linear maps:
In the process theory of **linear maps**:

(L1) Every type has a (finite) *basis*:

\[
\left( \text{for all } i : \begin{array}{c}
i \\
f \\
i \\
g \\
i \\
g \end{array} = \begin{array}{c}
i \\
f \\
i \\
g \end{array} \right) \implies \begin{array}{c}
f \\
f \end{array} = \begin{array}{c}
g \\
g \end{array}
\]
In the process theory of **linear maps**:

(L1) Every type has a (finite) **basis**:

\[
\left( \text{for all } i : \quad f_i = g_i \implies f = g \right)
\]

(L2) Processes can be **summed**:

\[
\sum_i f_i = \left( \sum_i h_i \right) f = \sum_i h_i f
\]
In the process theory of **linear maps**: 

**(L1)** Every type has a (finite) *basis*:

\[
\begin{align*}
\left( \text{for all } i : \begin{array}{c}
\sum_i f_i = \sum_i g_i \\
\sum_i f_i = \sum_i g_i
\end{array} \right) \implies \begin{array}{c}
\sum_i f_i = \sum_i g_i \\
\sum_i f_i = \sum_i g_i
\end{array}
\end{align*}
\]

**(L2)** Processes can be *summed*:

\[
\begin{align*}
\sum_i f_i = \sum_i (\sum_i h_i f_i) = \sum_i g_i
\end{align*}
\]

**(L3)** Numbers are the *complex numbers*: \( \Lambda \in \mathbb{C} \)
Bases $\Leftrightarrow$ process tomography

**Theorem**

\[
\left( \begin{array}{c}
\text{for all } \downarrow^i, \downarrow^j :
\begin{array}{c}
\begin{array}{c}
\uparrow^j \\
\downarrow^i
\end{array}
\end{array}
= \\
\begin{array}{c}
\begin{array}{c}
\uparrow^j \\
\downarrow^i
\end{array}
\end{array}
\end{array} \right) \implies f = g
\]
Bases ⇔ process tomography

Theorem

\[
\begin{pmatrix}
\text{for all } i, j : f_i^j = g_i^j \\
\end{pmatrix} \implies f_i^j = g_i^j
\]

Proof.
Bases $\Leftrightarrow$ process tomography

**Theorem**

\[
\left( \text{for all } i, j : \begin{array}{c}
\begin{array}{c}
\text{j} \\
\text{f}
\end{array} \\
\begin{array}{c}
\text{i} \\
\text{j}
\end{array}
\end{array} = \begin{array}{c}
\begin{array}{c}
\text{j} \\
\text{g}
\end{array} \\
\begin{array}{c}
\text{i} \\
\text{j}
\end{array}
\end{array} \right) \implies \begin{array}{c}
\begin{array}{c}
\text{f} \\
\text{i}
\end{array} = \begin{array}{c}
\begin{array}{c}
\text{g} \\
\text{i}
\end{array}
\end{array}
\end{array}
\]

**Proof.**

\[
\begin{array}{c}
\begin{array}{c}
\text{j} \\
\text{f}
\end{array} = \begin{array}{c}
\begin{array}{c}
\text{j} \\
\text{g}
\end{array}
\end{array}
\end{array}
\]
Bases ⇔ process tomography

Theorem

\[
\begin{pmatrix}
\text{for all } i, j : \quad f_i = g_i \\
\end{pmatrix}
\implies f = g
\]

Proof.
Bases $\Leftrightarrow$ process tomography

Theorem

\[
\begin{cases}
\text{for all } i, j : f_i = g_i \Rightarrow f_j = g_j
\end{cases}
\]

Proof.

\[
f = g
\]
Bases $\Leftrightarrow$ process tomography

Theorem

$$\begin{cases}
\text{for all } i, j : \begin{array}{c}
\downarrow j \\
\downarrow i \\
\uparrow f \\
\uparrow g \\
\end{array} = \begin{array}{c}
\downarrow j \\
\downarrow i \\
\uparrow f \\
\uparrow g \\
\end{array} \Rightarrow \begin{array}{c}
\uparrow f \\
\uparrow g \\
\end{array} = \begin{array}{c}
\uparrow f \\
\uparrow g \\
\end{array}
\end{cases}$$

Proof.

$$\begin{array}{c}
\uparrow f \\
\uparrow g \\
\end{array} = \begin{array}{c}
\uparrow f \\
\uparrow g \\
\end{array}$$
Bases ⇔ process tomography

Theorem

\[ \begin{bmatrix}
  f_{1j} & f_{2j} & \cdots & f_{mj} \\
  f_{1j} & f_{2j} & \cdots & f_{mj} \\
  \vdots & \vdots & \ddots & \vdots \\
  f_{1i} & f_{2i} & \cdots & f_{mi}
\end{bmatrix}
\]

for all \( i, j \), \( f = g \) \( \Rightarrow \) \( f = g \)

- In other words, \( f \) is uniquely fixed by its *matrix*:

\[ f_i^j :=
\]

where \( f_i^j :=
\]
What about the Born rule?
The Born rule for relations

\[ \text{test} \left\{ \begin{array}{c} \phi \\ \psi \end{array} \right\} \quad \mathbb{B} := \{0, 1\} \]
The Born rule for relations

\[ \Psi : \{ 0, 1 \} \]

possibility
The Born rule for **linear maps**

![Diagram](image)
Fixing the problem

\[
\begin{align*}
\text{test} & \quad \{ \phi, \phi \} \\
\text{state} & \quad \{ \psi, \psi \}
\end{align*}
\]

\[\mathbb{R}_{\geq 0}\] probability
Doubled states and effects

Letting:

\[ \hat{\psi} := \begin{array}{c} \psi \\
\psi
\end{array} \quad \text{and} \quad \hat{\phi} := \begin{array}{c} \phi \\
\phi
\end{array} \]
Doubled states and effects

Letting:

\[
\psi := \begin{array}{c}
\psi \\
\end{array}
\quad \text{and} \quad
\phi := \begin{array}{c}
\phi \\
\end{array}
\]

yields...

\[
\begin{aligned}
\text{test} & \{ \psi := \begin{array}{c}
\psi \\
\phi \\
\end{array} \} \\
\text{state} & \{ \psi := \begin{array}{c}
\psi \\
\phi \\
\end{array} \}
\end{aligned}
\]

\[
\begin{aligned}
\text{probability} & = \begin{array}{c}
\phi \\
\phi \\
\end{array}
\end{aligned}
\]
A new process theory from an old one...

- The theory of **pure quantum maps** has types:

```
\begin{array}{c}
\vdash \\
\hat{A}
\end{array}
```
A new process theory from an old one...

- The theory of **pure quantum maps** has types:

  \[
  \widehat{\mathcal{A}} := \mathcal{A}
  \]

- and processes:

  \[
  \widehat{f} = f \quad \text{for all processes } f \text{ from linear maps.}
  \]
Embedding the old theory

- **linear maps** embed in **quantum maps**, and this embedding preserves diagrams:

\[
\begin{pmatrix}
\begin{array}{c}
\hat{g} \\
\hat{h}
\end{array}
\end{pmatrix}
\begin{pmatrix}
\begin{array}{c}
\hat{f} \\
\hat{h}
\end{array}
\end{pmatrix}
\begin{pmatrix}
\begin{array}{c}
g \\
h
\end{array}
\end{pmatrix}
\]
Embedding the old theory

- **linear maps** embed in **quantum maps**, and this embedding preserves diagrams:

  \[
  \begin{pmatrix}
  f \\
  g \\
  h
  \end{pmatrix}
  =
  \begin{pmatrix}
  \hat{f} \\
  \hat{g} \\
  \hat{h}
  \end{pmatrix}
  \]

- But now we’re in a bigger space, so there is room for something new
Embedding the old theory

- **linear maps** embed in **quantum maps**, and this embedding preserves diagrams:

  \[
  \begin{pmatrix}
  f \\
  g \\
  h
  \end{pmatrix}
  =
  \begin{pmatrix}
  \hat{g} \\
  \hat{h} \\
  \hat{f}
  \end{pmatrix}
  \]

- But now we’re in a bigger space, so there is room for something new, *discarding*:

  \[
  \begin{array}{c}
  \hat{\psi} \\
  \end{array}
  =
  \quad
  \]

  \[
  \begin{array}{c}
  \psi \\
  \end{array}
  \]
What is discarding?

\[
\hat{\psi} = \hat{\psi} \hat{\psi}
\]

So discarding is defined as the effect:

\[
\hat{\psi} = d \hat{\psi} = \Rightarrow d
\]

In fact, this is the unique map with this property. Let \( \{ \hat{\psi}_i \} \) be a basis of pure states (e.g. \( z^+ \), \( z^- \), \( x^+ \), \( y^+ \)), then:

\[
\hat{\psi}_i = d \hat{\psi}_i = \Rightarrow d
\]
What is discarding?

\[ \hat{\psi} = \psi \cdot \psi \]

So discarding is defined as the effect:

\[ \mathbf{d} = \hat{\psi} \psi \]

In fact, this is the unique map with this property. Let \( \{ \hat{\psi}_i \} \) be a basis of pure states (e.g. \( z^+ \), \( z^- \), \( x^+ \), \( y^+ \)), then:

\[ \hat{\psi}_i = d \hat{\psi}_i = \mathbf{d} \]
What is discarding?

\[ \hat{\psi} = \psi \]

So discarding is defined as the effect:

\[ \Rightarrow \]

In fact, this is the unique map with this property. Let \( \{\hat{\psi}_i\} \) be a basis of pure states (e.g. \( z^+ \), \( z^- \), \( x^+ \), \( y^+ \)), then:

\[ \hat{\psi}_i = d \hat{\psi}_i = \Rightarrow = d \]
What is discarding?

\[ \hat{\psi} = \psi \]

So discarding is defined as the effect:

\[ \hat{\text{discarding}} := \]
What is discarding?

\[ \hat{\psi} = \hat{\psi} = \hat{\psi} = \bullet \]

- So discarding is defined as the effect:

\[ := \]

- In fact, this is the unique map with this property. Let \( \{ \hat{\psi}_i \}_i \) be a basis of pure states (e.g. \( z^+, z^-, x^+, y^+ \)), then:

\[ \hat{\psi}_i := \]

\[ \hat{\psi}_i = \bullet = \hat{\psi}_i \]
What is discarding?

- So discarding is defined as the effect:

\[ \overline{\psi} = \begin{array}{c} \overline{\psi} \\ \underline{\psi} \end{array} := \begin{array}{c} \vdots \\ \vdots \end{array} \]

- In fact, this is the unique map with this property. Let \( \{\hat{\psi}_i\}_i \) be a basis of pure states (e.g. \( z^+, z^-, x^+, y^+ \)), then:

\[ \overline{\hat{\psi}_i} = \begin{array}{c} \vdots \\ \vdots \end{array} = \begin{array}{c} \hat{d} \\ \hat{\psi}_i \end{array} \implies \overline{\hat{\psi}_i} = \begin{array}{c} \hat{d} \\ \hat{\psi}_i \end{array} \]
Definition
The process theory of **quantum maps** consists of all processes obtained from pure quantum maps and discarding:
Definition

The process theory of quantum maps consists of all processes obtained from pure quantum maps and discarding:

\[
\begin{align*}
\{ \hat{f} \} & \quad \dashv \quad \\
\quad \cdots
\end{align*}
\]

- e.g. \( \rho := \hat{\psi} \) and \( \Phi := \hat{f} \circ \hat{h} \)
Causality

- This gives all quantum processes, including post-selected ones
This gives all quantum processes, including post-selected ones.

To get all of the *deterministically realisable* processes, we additionally require *causality*:

\[ \Phi = \overline{\overline{T}} \]
• This gives all quantum processes, including post-selected ones
• To get all of the *deterministically realisable* processes, we additionally require *causality*:

\[ \Phi = \tilde{\Phi} \]

• Causality \( \implies \) no-signalling:
This gives all quantum processes, including post-selected ones.

To get all of the deterministically realisable processes, we additionally require causality:

\[ \Phi = \Gamma \]

Causality \implies \text{no-signalling}:

\[ \sigma \]
Causality

- This gives all quantum processes, including post-selected ones.
- To get all of the *deterministically realisable* processes, we additionally require *causality*:

\[
\Phi = \quad =
\]

- Causality \(\implies\) no-signalling:

\[
\Phi \quad =
\]

\[
\Psi
\]

Alice \quad Bob

\[
\Phi
\]

\[
\Psi
\]

Alice \quad Bob
Causality

- This gives all quantum processes, including post-selected ones.
- To get all of the *deterministically realisable* processes, we additionally require *causality*:

\[
\Phi = \Psi
\]

- Causality $\implies$ no-signalling:

\[\Phi \rho \Psi \mathrel{=} \Psi \rho \Psi \mathrel{=} \Psi'\]
Purification

- Any quantum map extends to a pure quantum map on an extended system:

\[
\begin{array}{c}
\Phi \\
\end{array}
= 
\begin{array}{c}
\hat{f}
\end{array}
\]
Purification

- Any quantum map extends to a pure quantum map on an extended system:

\[ \Phi = \hat{f} \]

- This is built-in to our definition of quantum maps:
Purification

• Any quantum map extends to a pure quantum map on an extended system:

\[ \Phi = \hat{f} \]

• This is built-in to our definition of quantum maps:

• If \( \Psi \) causal, \( \hat{f} \) must be isometry: Stinespring dilation.
No-broadcasting from pure extension

**Theorem**

A state is pure if and only if any *extension* separates:

\[
\hat{\psi} = \rho \quad \Rightarrow \quad \rho = \hat{\psi} \quad \hat{\psi} = \rho'
\]
No-broadcasting from pure extension

**Theorem**
A state is pure if and only if any *extension* separates:

\[\hat{\psi} = \rho \implies \rho = \hat{\psi}\rho'\]

**Corollary**
There exists no quantum map \(\Delta\) such that:

\[\Delta = \Delta = \Delta\]
No-broadcasting from pure extension - proof

Broadcast to the left:

\[
\begin{align*}
\Delta & \rightarrow \\
\end{align*}
\]
No-broadcasting from pure extension - proof

Broadcast to the left:

\[ \Delta = \]

Bend the wire:

\[ \Delta = \]
No-broadcasting from pure extension - proof

Broadcast to the left:

\[
\begin{align*}
\Delta & = \quad \quad \quad \quad \\
\end{align*}
\]

Bend the wire:

\[
\begin{align*}
\Delta & = \quad \quad \quad \quad \\
\end{align*}
\]

\[
\begin{align*}
\Delta & = \quad \quad \quad \quad \\
\end{align*}
\]
No-broadcasting from pure extension - proof

Broadcast to the left:

\[ \Delta = \]

Bend the wire:

\[ \Delta = \rho \]

Unbend the wire and try to broadcast to the right:

\[ \Delta = \rho \]
Classical systems

- Protocols, experiments, etc. are always about the interaction of quantum and classical systems
Classical systems

- Protocols, experiments, etc. are always about the interaction of quantum and classical systems
- We extend graphical language:
  
  quantum systems $\rightarrow$ double wires

  classical systems $\rightarrow$ single wires
Classical systems

- Protocols, experiments, etc. are always about the interaction of quantum and classical systems
- We extend graphical language:
  
  quantum systems $\rightarrow$ double wires

  classical systems $\rightarrow$ single wires

- States are *probability distributions*:

\[
p = \sum_j p^j \quad \leftrightarrow \quad \begin{pmatrix} p^1 \\ p^2 \\ \vdots \\ p^n \end{pmatrix}
\]
Classical systems

- Protocols, experiments, etc. are always about the interaction of quantum and classical systems
- We extend graphical language:
  - quantum systems → double wires
  - classical systems → single wires

- States are *probability distributions*:

\[
p = \sum_j p_j
\]

- Processes are *stochastic maps*:

\[
f \leftrightarrow \begin{pmatrix} p_1^1 & p_1^2 & \cdots & p_m^1 \\ p_1^2 & p_2^2 & \cdots & p_m^2 \\ \vdots & \vdots & \ddots & \vdots \\ p_1^n & p_2^n & \cdots & p_m^n \end{pmatrix}
\]
Classical operations

- Deleting is marginalisation:

\[ \Idem := \sum_i i \]
Classical operations

• Deleting is marginalisation:

\[ \bigcirc := \sum_i \bigtriangleup_i \]

• Classical causality just means stochastic:

\[ f = \bigcirc \]
Classical operations

- Deleting is marginalisation:

\[ i := \sum_i \]

- Classical causality just means stochastic:

\[ f = \]

- We *can* broadcast classically:

\[ = \quad = \quad \text{where} \quad := \sum_i \]
Generalising to *spiders*

- These generalise to a whole family of maps, called *spiders*:
Generalising to *spiders*

- These generalise to a whole family of maps, called *spiders*:

\[
\begin{align*}
\overbrace{\begin{array}{c}
\vdots \\
\vdots \\
\vdots
\end{array}}^{n} & \quad \overbrace{\begin{array}{c}
\vdots \\
\vdots \\
\vdots
\end{array}}^{m} \\
\text{:=} & \quad \sum_{i} \overbrace{\begin{array}{c}
\vdots \\
\vdots \\
\vdots
\end{array}}^{n} \\
\overbrace{\begin{array}{c}
\vdots \\
\vdots \\
\vdots
\end{array}}^{i} & \quad \overbrace{\begin{array}{c}
\vdots \\
\vdots \\
\vdots
\end{array}}^{i} \\
\overbrace{\begin{array}{c}
\vdots \\
\vdots \\
\vdots
\end{array}}^{\cdots} & \quad \overbrace{\begin{array}{c}
\vdots \\
\vdots \\
\vdots
\end{array}}^{i} \\
\overbrace{\begin{array}{c}
\vdots \\
\vdots \\
\vdots
\end{array}}^{i} & \quad \overbrace{\begin{array}{c}
\vdots \\
\vdots \\
\vdots
\end{array}}^{i}
\end{align*}
\]

- where the only rule to remember is:

\[
\begin{align*}
\overbrace{\begin{array}{c}
\vdots \\
\vdots \\
\vdots
\end{array}}^{\cdots} & \quad \overbrace{\begin{array}{c}
\vdots \\
\vdots \\
\vdots
\end{array}}^{\cdots} \\
\text{:=} & \quad \overbrace{\begin{array}{c}
\vdots \\
\vdots \\
\vdots
\end{array}}^{\cdots}
\end{align*}
\]
Quantum spiders

• A quantum spider is a classical spider, doubled:

\[
\begin{array}{c}
\cdots \\
\cdots \\
\end{array} \\
\begin{array}{c}
\cdots \\
\end{array}
\]

\[\begin{array}{c}
\text{double}
\begin{array}{c}
\cdots \\
\end{array}
\end{array} =
\begin{array}{c}
\cdots \\
\end{array}
\begin{array}{c}
\cdots \\
\end{array}
\]

\[
\begin{array}{c}
\cdots \\
\cdots \\
\end{array} \\
\begin{array}{c}
\cdots \\
\cdots \\
\end{array}
\]
Quantum spiders

- A quantum spider is a classical spider, doubled:

  $\ldots$ := double $\left( \begin{array}{c} \ldots \\ \ldots \end{array} \right) = \ldots$

- An example is the GHZ state:

  $\text{GHZ} := \begin{array}{c} \text{GHZ} \end{array} = \text{double} \left( \sum_i \begin{array}{c} i \\ i \\ i \\ i \end{array} \right)$
Quantum spiders

- A quantum spider is a classical spider, doubled:

\[ \text{\ldots} \quad := \quad \text{double} \left( \begin{array}{c} \text{\ldots} \\ \text{\ldots} \end{array} \right) \quad = \quad \ldots \]

- An example is the GHZ state:

\[ \text{GHZ} \quad := \quad \text{double} \left( \sum_i i \quad \begin{array}{c} \text{i} \\ \text{i} \\ \text{i} \end{array} \right) \quad = \quad \text{\ldots} \]

- They also fuse:
Bastard spiders

- The third type of spider treats some legs as classical, and some pairs of legs as quantum:

![Diagram of bastard spiders]
Bastard spiders

- The third type of spider treats some legs as classical, and some pairs of legs as quantum:

\[
\begin{array}{c}
\vdots \\
\vdots \\
\vdots \\
\vdots \\
\end{array}
\quad :=
\begin{array}{c}
\vdots \\
\vdots \\
\vdots \\
\vdots \\
\end{array}
\]

- We call these (seemingly) weird things \textit{bastard spiders}
• The third type of spider treats some legs as classical, and some pairs of legs as quantum:

\[
\begin{array}{c}
\cdots \cdots \\
\vdots \\
\cdots \cdots \\
\end{array}
\quad :=

\begin{array}{c}
\cdots \cdots \\
\cdots \cdots \\
\cdots \cdots \\
\end{array}
\]

• We call these (seemingly) weird things \textit{bastard spiders}

• Again they fuse together:
Measurement’s a bastard

- The most important example is *ONB-measurement*:

\[
\begin{pmatrix}
P(1|\rho) \\
P(2|\rho) \\
\vdots \\
P(n|\rho)
\end{pmatrix}
\]
Measurement’s a bastard

• The most important example is *ONB-measurement*:

$$\rho \mapsto \begin{pmatrix} P(1|\rho) \\ P(2|\rho) \\ \vdots \\ P(n|\rho) \end{pmatrix}$$

• whose adjoint is *ONB-encoding*:

$$i \mapsto i$$
The most important example is \textit{ONB-measurement}:

\[
\rho \mapsto \begin{pmatrix}
P(1|\rho) \\
P(2|\rho) \\
\vdots \\
P(n|\rho)
\end{pmatrix}
\]

whose adjoint is \textit{ONB-encoding}:

\[
\begin{array}{l}
\phi \mapsto i \\
\end{array}
\]

Combining these yields more general stuff, e.g. non-demo measurements:
Multi-coloured spiders

- Different bases $\rightarrow$ different coloured spiders

\[
m \mathcal{S} = \sum_i \left( i \mathcal{S}_i \right) = \sum_i \left( i \mathcal{S}_i \right) \quad \text{for different bases} \]

Two spiders are complementary if:

\[
\text{encode in} \quad \text{+} \quad \text{measure in} \quad = \quad \text{no data transfer}
\]
Multi-coloured spiders

- Different bases $\rightarrow$ different coloured spiders

- Two spiders $\circ$ and $\bullet$ are complementary if:

  $\circ = \bullet$

  $(\text{encode in } \circ) + (\text{measure in } \bullet) = (\text{no data transfer})$
Complementarity – Stern-Gerlach

- For example, we can model Stern-Gerlach:
Complementarity – Stern-Gerlach

- For example, we can model Stern-Gerlach:

\[
\begin{array}{c}
  \text{S} \\
  \text{N} \\
  \text{S} \\
  \text{N} \\
  \Rightarrow \text{X}
\end{array}
\]

which simplifies as:

\[
\begin{array}{c}
  \text{Z} \\
  \text{X} \\
  \text{Z} \\
  \Rightarrow \text{no data transfer}
\end{array}
\]
Applications

Picturing Quantum Processes
the rest...
Applications

Picturing Quantum Processes
the rest...

1. Quantum info: Complementarity and cousin strong complementary give graphical presentations for many protocols, e.g. QKD, QSS
Applications

Picturing Quantum Processes
the rest...

1. **Quantum info:** Complementarity and cousin *strong complementary* give graphical presentations for many protocols, e.g. QKD, QSS

2. **Quantum computing:** Complementary spiders give a handy toolkit for building quantum circuits and MBQC
Applications

Picturing Quantum Processes
the rest...

1. **Quantum info**: Complementarity and cousin *strong complementary* give graphical presentations for many protocols, e.g. QKD, QSS

2. **Quantum computing**: Complementary spiders give a handy toolkit for building quantum circuits and MBQC

3. **Quantum resources**: Framework for resource theories, e.g. entanglement, purity
Applications

Picturing Quantum Processes

the rest...

1. **Quantum info**: Complementarity and cousin *strong complementary* give graphical presentations for many protocols, e.g. QKD, QSS
2. **Quantum computing**: Complementary spiders give a handy toolkit for building quantum circuits and MBQC
3. **Quantum resources**: Framework for resource theories, e.g. entanglement, purity
4. **Quantum foundations**: (spoiler alert!) GHZ/Mermin argument in diagrams
Applications

Picturing Quantum Processes
the rest...

1. **Quantum info:** Complementarity and cousin *strong complementary* give graphical presentations for many protocols, e.g. QKD, QSS
2. **Quantum computing:** Complementary spiders give a handy toolkit for building quantum circuits and MBQC
3. **Quantum resources:** Framework for resource theories, e.g. entanglement, purity
4. **Quantum foundations:** (spoiler alert!) GHZ/Mermin argument in diagrams

Thanks!