Learning to predict arbitrary quantum processes

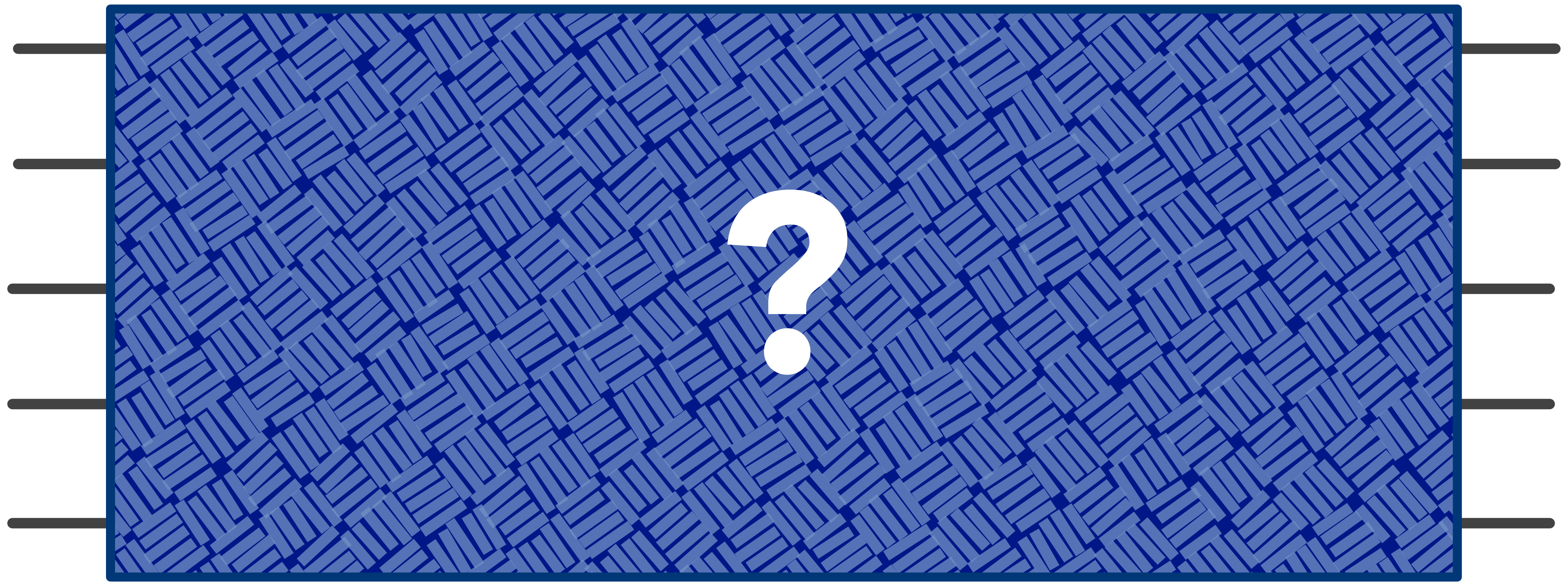
Presenter: Hsin-Yuan Huang (Robert)
Joint work with Sitan Chen and John Preskill

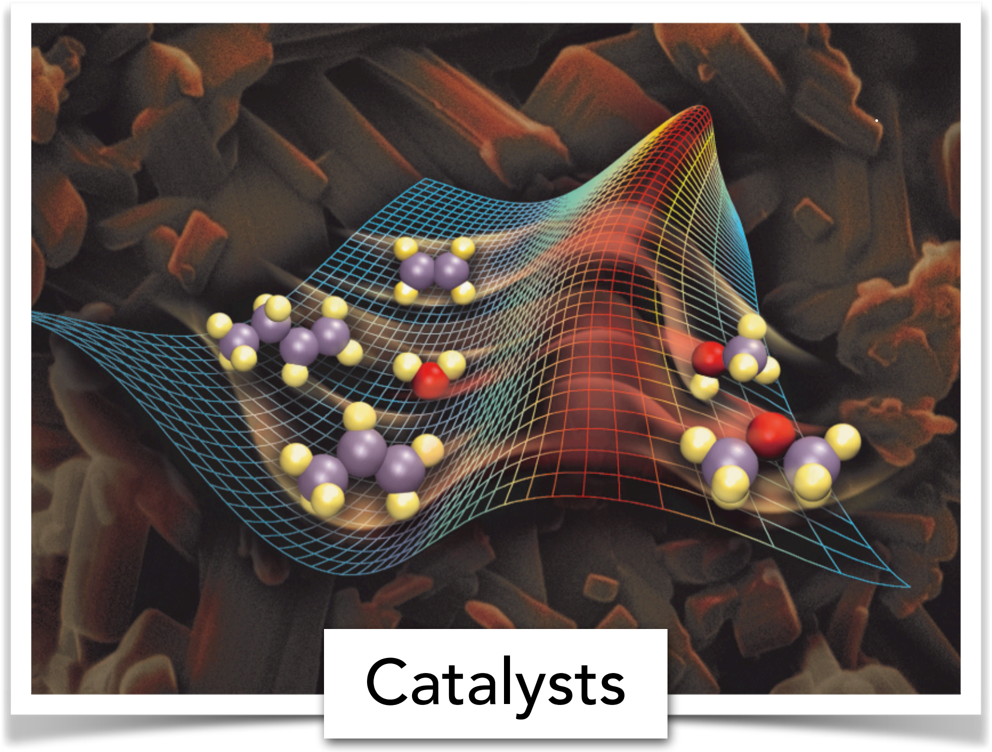
IQIM
INSTITUTE FOR QUANTUM INFORMATION AND MATTER
Caltech
HARVARD UNIVERSITY
aws

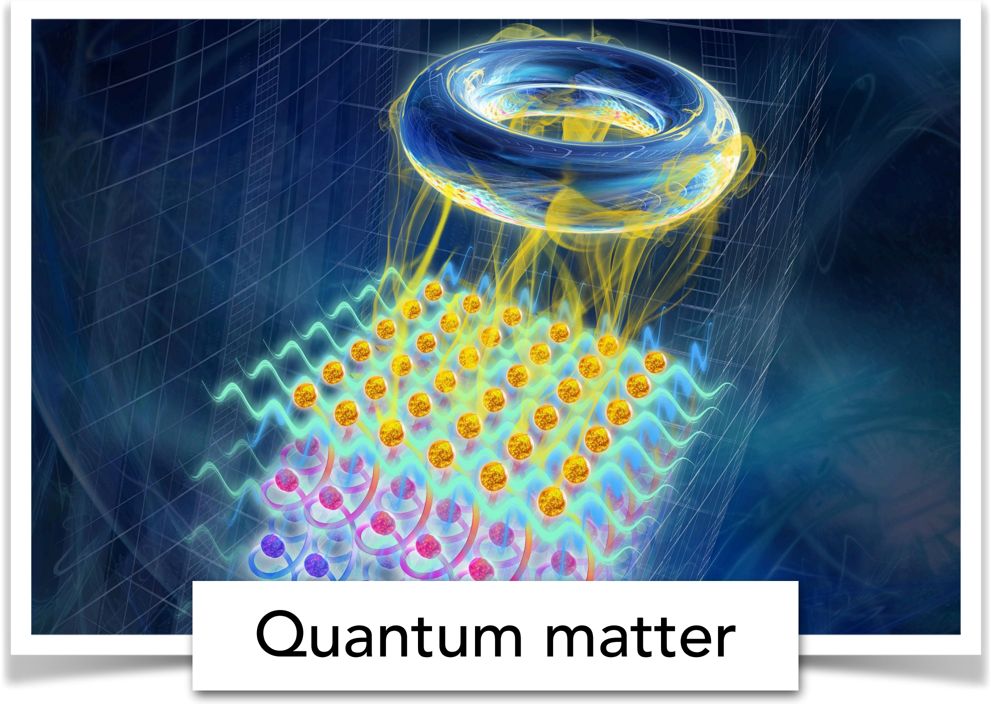
Motivation

- We have seen substantial recent progress on efficiently learning to predict quantum states.
- Are there efficient algorithms for learning to predict quantum circuits / processes?

A high-complexity quantum process







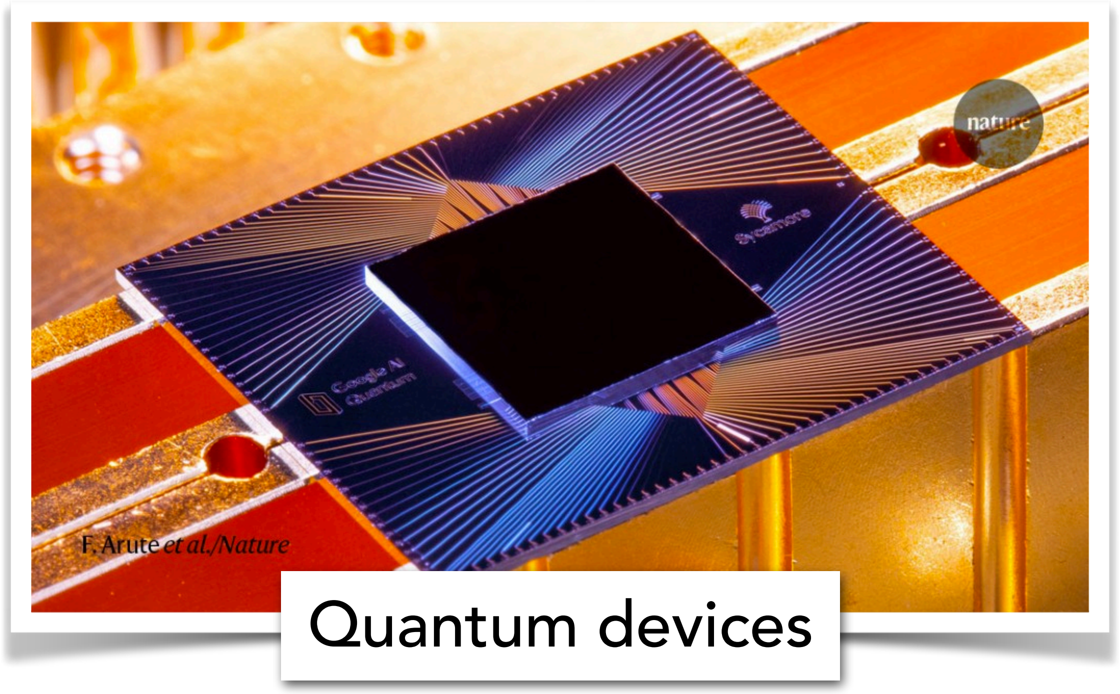


Image credits: (Top left) https://www.energy.gov/science/doe-explainscatalysts (Top right) https://theconversation.com/as-pharmaceutical-use-continues-to-rise-side-effects-are-becoming-a-costly-health-issue-105494 (Bottom left) https://news.mit.edu/2019/ultra-quantum-matter-uqm-research-given-8m-boost-0529 (Bottom right) https://www.nature.com/articles/d41586-019-03213-z

The Setting

- In this work, we focus on training an ML model to learn and predict

$$\rho, O \mapsto f_{\mathcal{E}}(\rho, O) = \mathrm{Tr}(O\mathcal{E}(\rho)),$$

  where ρ is an input quantum state, $\mathcal{E}$ is an (unknown) CPTP map, and O is an observable.

- This includes any function computable by a quantum computer (in exponential time).

The Setting

- In this work, we focus on training an ML model to learn and predict

$$\rho, O \mapsto f_{\mathscr{E}}(\rho, O) = \mathrm{Tr}(O\mathscr{E}(\rho)),$$

where ρ is an input quantum state, $\mathscr{E}$ is an (unknown) CPTP map, and O is an observable.

- This includes any function computable by a quantum computer (in exponential time).

Example 1

Predicting outcomes of physical experiments

ρ : initial state given by classical input x

$\mathscr{E}$: the physical process in the experiment

O : what the scientist measure

The Setting

- In this work, we focus on training an ML model to learn and predict

$$\rho, O \mapsto f_{\mathscr{E}}(\rho, O) = \mathrm{Tr}(O\mathscr{E}(\rho)),$$

where ρ is an input quantum state, $\mathscr{E}$ is an (unknown) CPTP map, and O is an observable.

- This includes any function computable by a quantum computer (in exponential time).

Example 1

Predicting outcomes of physical experiments

ρ : initial state given by classical input x

$\mathscr{E}$: the physical process in the experiment

O : what the scientist measure

Example 2

Training quantum neural networks

ρ : input state encoding classical input x

$\mathscr{E}$: the quantum neural network to learn

O : a single fixed observable

The Setting

- In this work, we focus on training an ML model to learn and predict

$$\rho, O \mapsto f_{\mathscr{E}}(\rho, O) = \mathrm{Tr}(O\mathscr{E}(\rho)),$$

where ρ is an input quantum state, $\mathscr{E}$ is an (unknown) CPTP map, and O is an observable.

- This includes any function computable by a quantum computer (in exponential time).

Example 1

Predicting outcomes of physical experiments

ρ : initial state given by classical input x

$\mathscr{E}$: the physical process in the experiment

O : what the scientist measure

Example 2

Training quantum neural networks

ρ : input state encoding classical input x

$\mathscr{E}$: the quantum neural network to learn

O : a single fixed observable

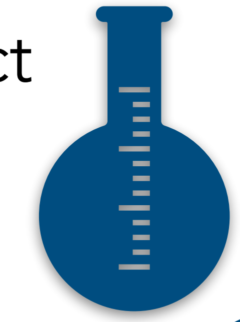
Example 3

Speeding up complex quantum dynamics

ρ : initial state of the physical system

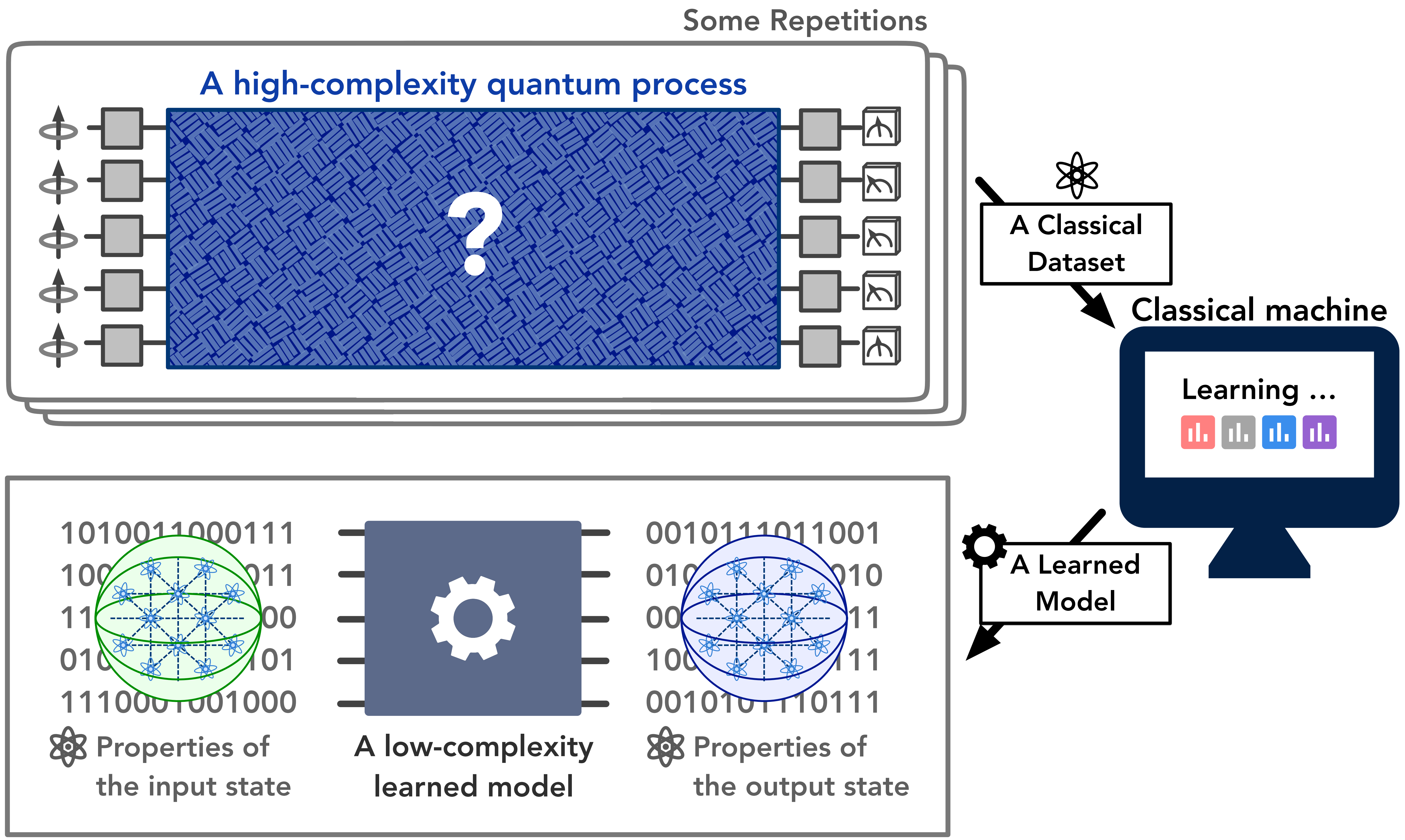
$\mathscr{E}$: the quantum dynamics to speed up

O : the property we want to predict



The goal of this work

Given an n-qubit CPTP map $\mathscr{E}$ that represents a high-complexity quantum process



Overview

- A classical version of the quantum problem
- A restricted version of the quantum problem
- Generalization to the original quantum problem

Overview

- A classical version of the quantum problem
- A restricted version of the quantum problem
- Generalization to the original quantum problem

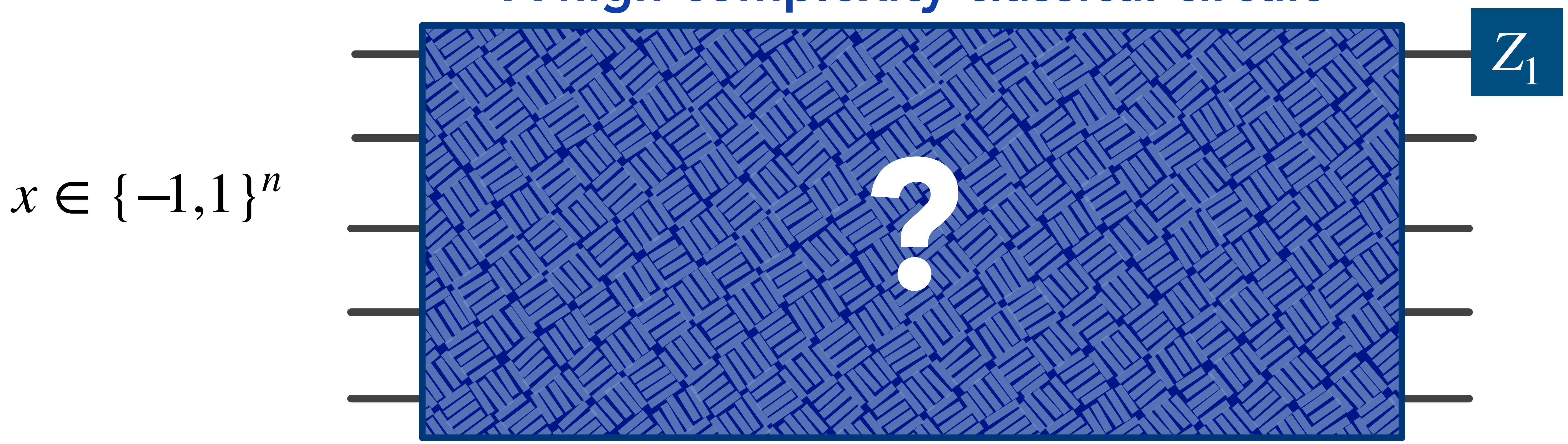
A Classical Problem

- Given an unknown classical Boolean circuit C mapping n bits to n bits.
- The input is now an n-bit string $x \in \{-1,1\}^n$.
- The 1st output bit of C for input x is equal to $f_C(x) = \mathrm{Tr}(Z_1 C(|x\rangle\langle x|))$.

Worst-case hardness

- The function f_C is equiv. to an exponentially long vector $\{-1,1\}^{2^n}$ with **no structure**.
- To learn a model $h(x)$, such that $\left|h(x) - f_C(x)\right|^2 < 0.5, \forall x \in \{-1,1\}^n$, we must query $f_C(x)$ for all input x. Query complexity: $\Theta(2^n)$.

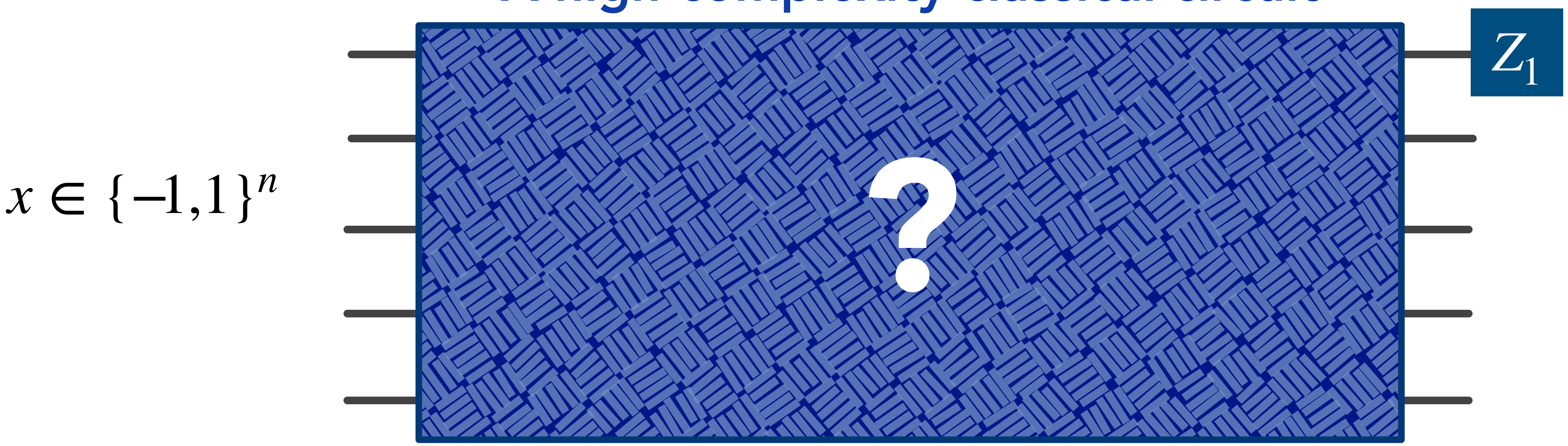
A high-complexity classical circuit



Average-case hardness

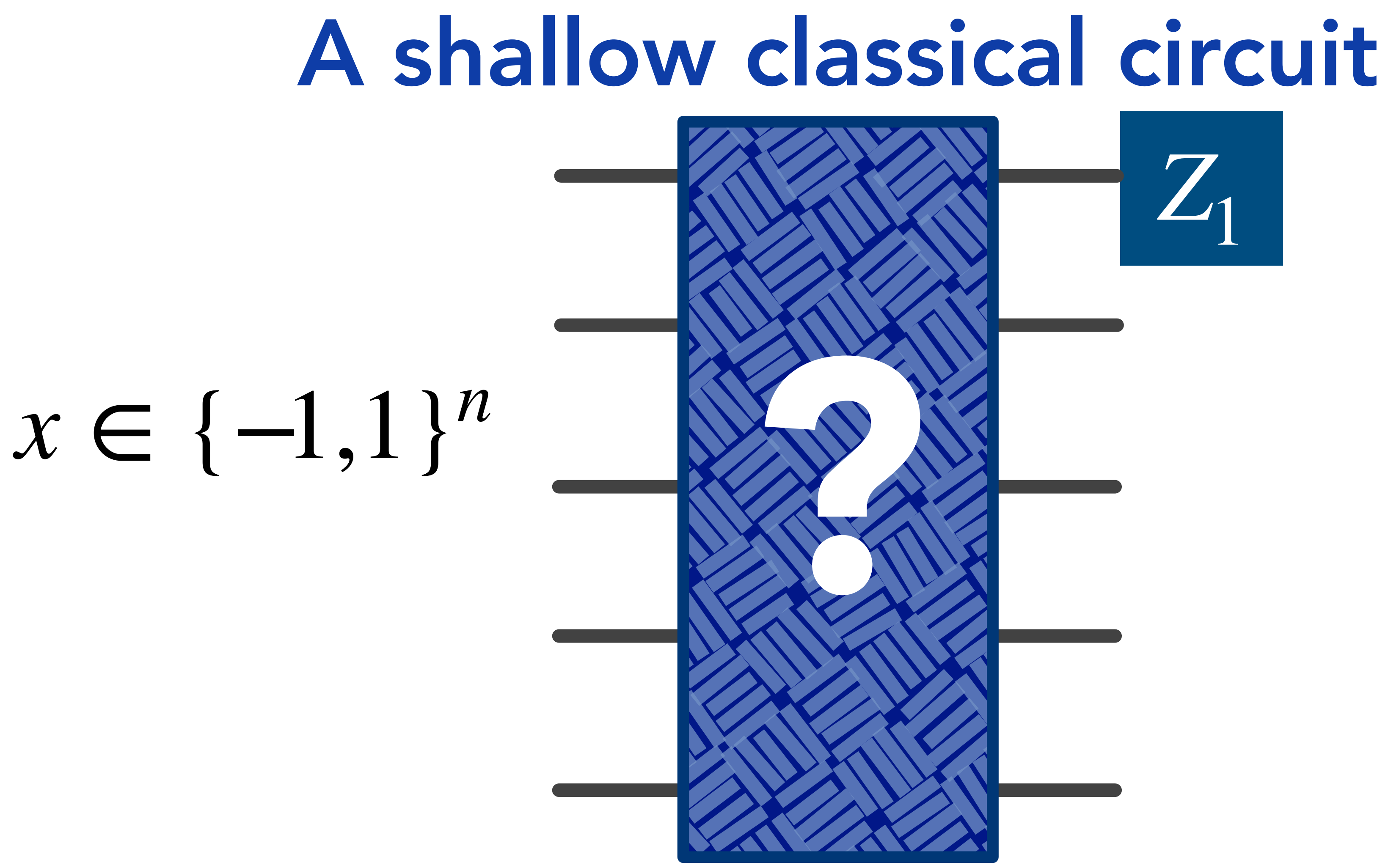
- The function f_C is equiv. to an exponentially long vector $\{-1,1\}^{2^n}$ with **no structure**.
- To learn a model $h(x)$, such that $\mathbb{E}_{x\sim\{-1,1\}^n} \left| h(x) - f_C(x) \right|^2 < 0.5$, we must query $f_C(x)$ for at least half of all x. Query complexity: $\Theta(2^n)$.

A high-complexity classical circuit



Average-case hardness for shallow classical circuits

- [AGS19] showed that learning $h(x)$, such that $\mathbb{E}_{x\sim\{-1,1\}^n} \left| h(x) - f_C(x) \right|^2 < 0.5$, is computationally hard (for both classical & quantum computers), even when the classical Boolean circuit is **constant-depth** (with majority gates, i.e., TC_0).



Overview

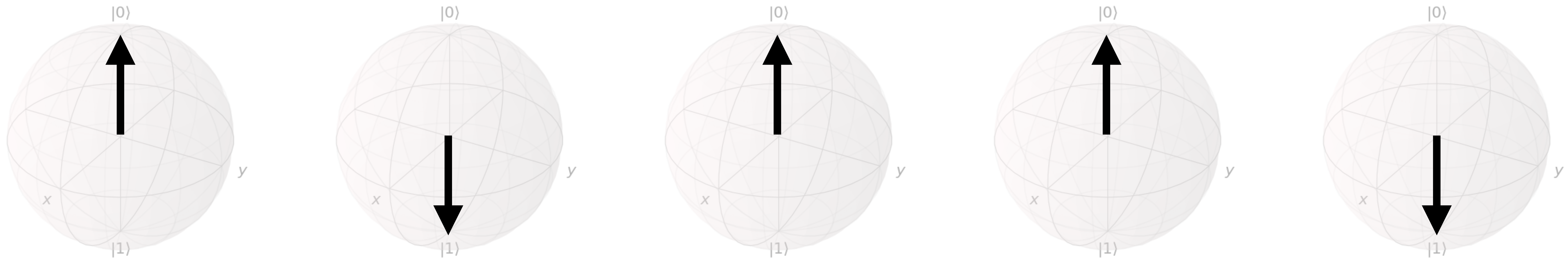
- A classical version of the quantum problem
- A restricted version of the quantum problem
- Generalization to the original quantum problem

Overview

- A classical version of the quantum problem
- A restricted version of the quantum problem
- Generalization to the original quantum problem

A Classical Problem

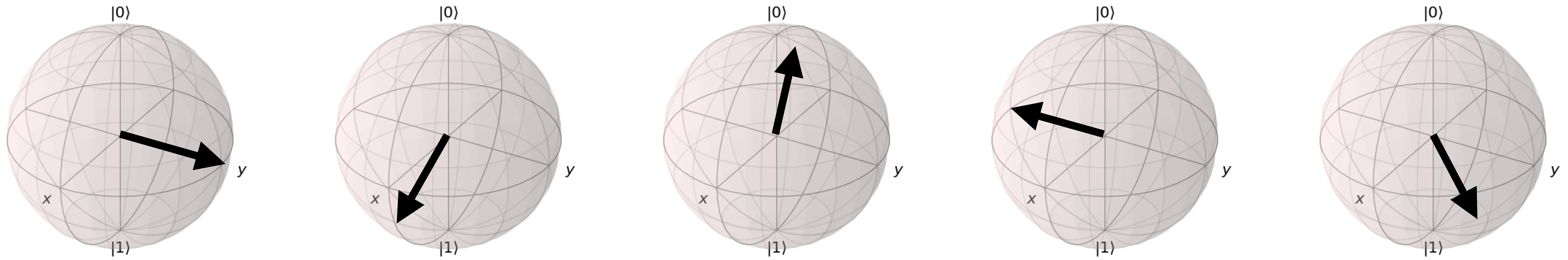
Input:



This is hard!

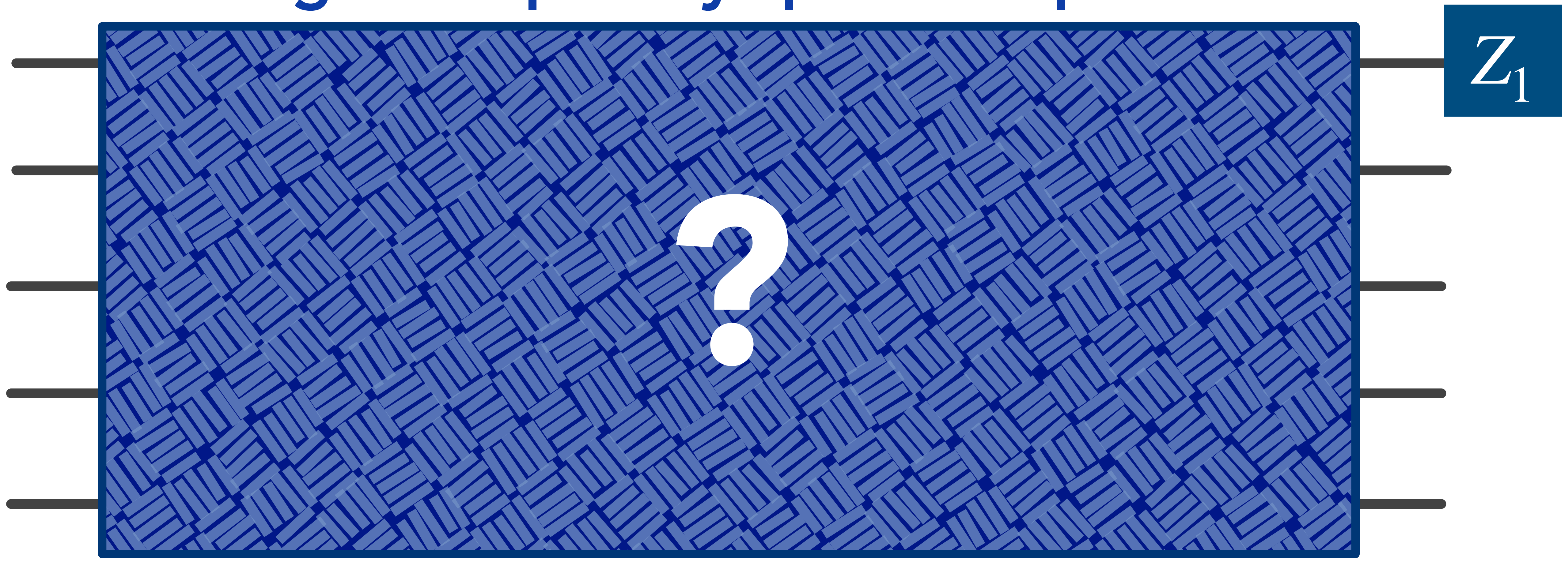
A Quantum Problem

Input:



$$\bigotimes_{i=1}^{n} |\psi_i\rangle \in (\mathbb{C}^2)^{\otimes n}$$

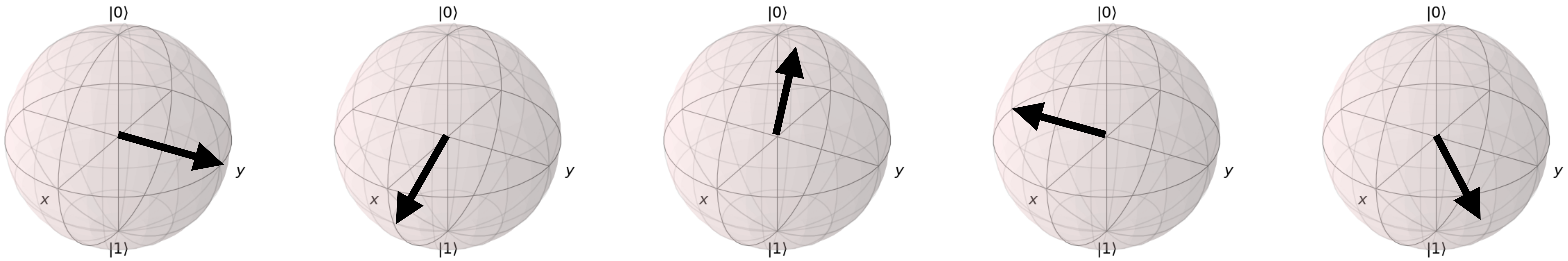
A high-complexity quantum process



Is this harder?

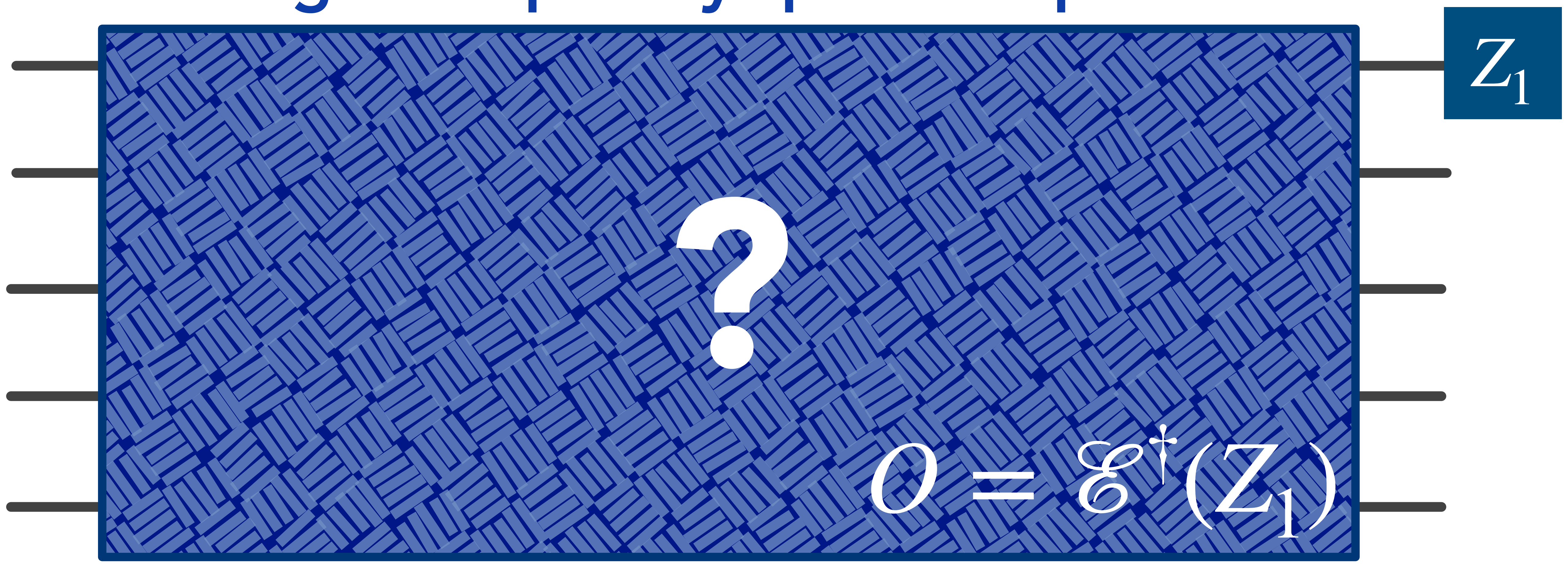
A Quantum Problem

Input:



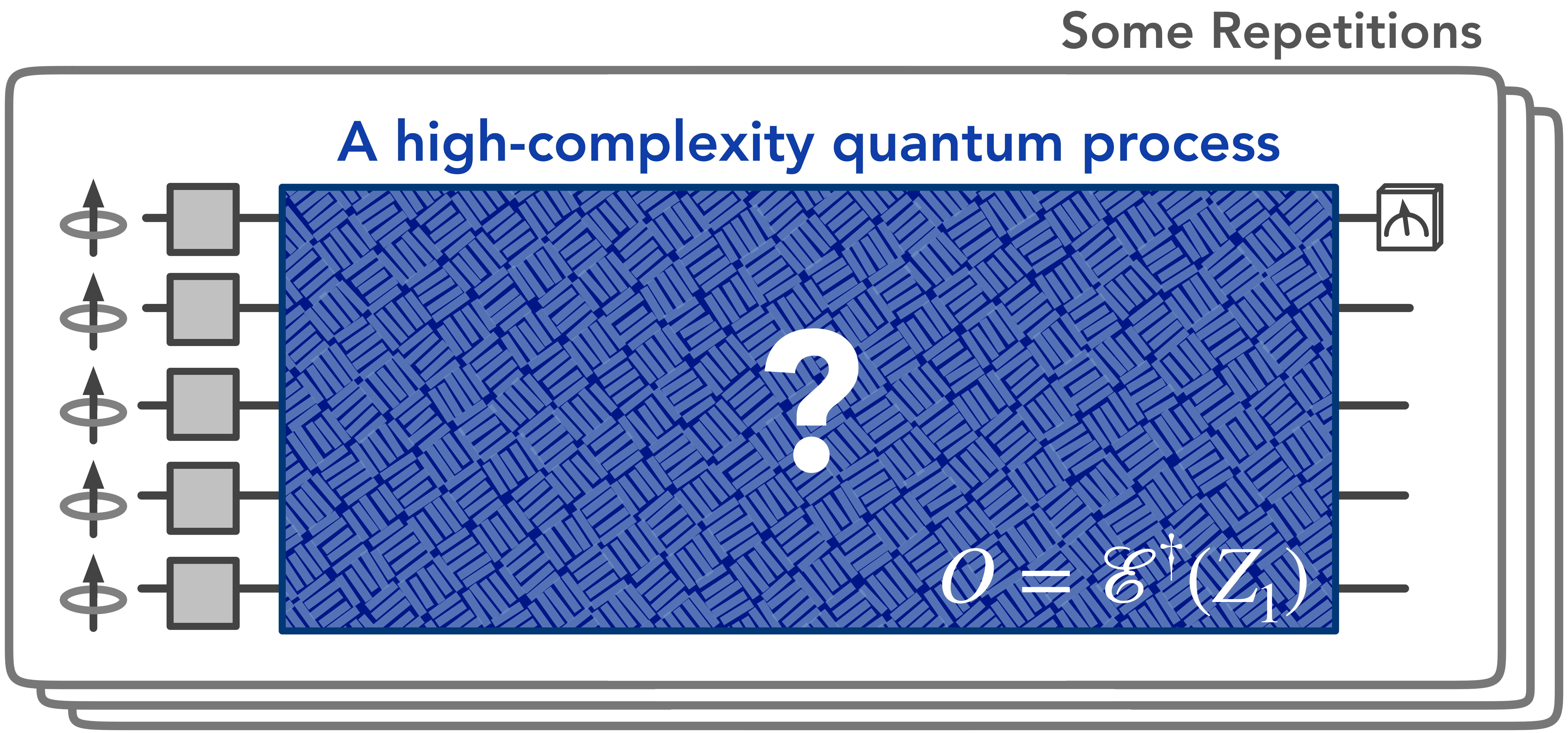
A high-complexity quantum process

$$\bigotimes_{i=1}^{n} |\psi_i\rangle \in (\mathbb{C}^2)^{\otimes n}$$



Is this harder?

A Classical Dataset



Each repetition prepares a random product state, and measures the 1st qubit in the Z basis

A Classical Dataset

Classical Dataset about O

$$|\psi_\ell\rangle = \bigotimes_{i=1}^{n} |\psi_{\ell,i}\rangle \mapsto y_\ell, \quad \mathbb{E}[y_\ell] = \langle \psi_\ell | O | \psi_\ell \rangle$$

$$\text{for } \ell = 1, \ldots, N.$$

Each repetition prepares a random product state, and measures the 1st qubit in the Z basis

The Prediction Task

Classical Dataset about O

$$|\psi_\ell\rangle = \bigotimes_{i=1}^{n} |\psi_{\ell,i}\rangle \mapsto y_\ell, \quad \mathbb{E}[y_\ell] = \langle \psi_\ell | O | \psi_\ell \rangle$$

for $\ell = 1, \ldots, N$.

Given a new state $|\psi\rangle = \bigotimes_{i=1}^{n} |\psi_i\rangle \in (\mathbb{C}^2)^{\otimes n}$,

how to predict $\langle \psi | O | \psi \rangle$ accurately?

Worst-case hardness

- To learn a model $h(|\psi\rangle)$, such that $\left| h(|\psi\rangle) - \langle\psi|O|\psi\rangle \right|^2 < 0.5, \forall |\psi\rangle = \bigotimes_{i=1}^{n} |\psi_i\rangle$, the problem is at least as hard as the classical problem.
- Hence, the query complexity is $\Omega(2^n)$.

Average-case hardness?

- To learn a model $h(|\psi\rangle)$, such that $\mathbb{E}_{|\psi\rangle=\bigotimes_{i=1}^{n}|\psi_i\rangle} \left| h(|\psi\rangle) - \langle\psi|O|\psi\rangle \right|^2 < 0.5$, is the problem still exponentially hard?

- Surprisingly, the answer is **no**. The problem can be done in quasi-polynomial time.

Low-weight approximation

$$O = \sum_{P \in \{I,X,Y,Z\}^{\otimes n}} \alpha_P P$$

$$O^{(\mathrm{low})} = \sum_{|P| \le k} \alpha_P P$$

Lemma (Low-weight approximation): $\mathbb{E}_{|\psi\rangle = \bigotimes_{i=1}^{n} |\psi_i\rangle} \left| \langle \psi | O | \psi \rangle - \langle \psi | O^{(\mathrm{low})} | \psi \rangle \right|^2 < \frac{1}{3^k}.$

Low-weight approximation

$$O = \sum_{P \in \{I,X,Y,Z\}^{\otimes n}} \alpha_P P$$

$$O^{(\mathrm{low})} = \sum_{|P| \le k} \alpha_P P$$

Lemma (Low-weight approximation): $\mathbb{E}_{|\psi\rangle = \bigotimes_{i=1}^{n} |\psi_i\rangle} \left| \langle \psi | O | \psi \rangle - \langle \psi | O^{(\mathrm{low})} | \psi \rangle \right|^2 < \frac{1}{3^k}.$

Interpretation: For most product state $|\psi\rangle = \bigotimes_{i=1}^{n} |\psi_i\rangle$, $\langle \psi | O | \psi \rangle \approx \langle \psi | O^{(\mathrm{low})} | \psi \rangle$.

Low-weight approximation

$$O = \sum_{P \in \{I,X,Y,Z\}^{\otimes n}} \alpha_P P$$

$$O^{(\mathrm{low})} = \sum_{|P| \le k} \alpha_P P$$

Lemma (Low-weight approximation): $\mathbb{E}_{|\psi\rangle = \bigotimes_{i=1}^{n} |\psi_i\rangle} \left| \langle \psi | O | \psi \rangle - \langle \psi | O^{(\mathrm{low})} | \psi \rangle \right|^2 < \frac{1}{3^k}.$

Interpretation: For most product state $|\psi\rangle = \bigotimes_{i=1}^{n} |\psi_i\rangle$, $\langle \psi | O | \psi \rangle \approx \langle \psi | O^{(\mathrm{low})} | \psi \rangle$.

Low-weight approximation **does not** hold in the classical version of this problem

Low-weight approximation

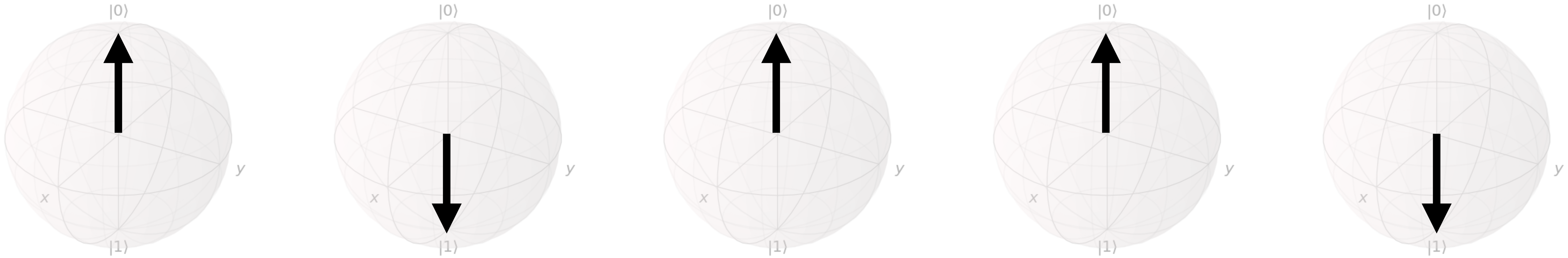
$$O = \sum_{P \in \{I,X,Y,Z\}^{\otimes n}} \alpha_P P$$

$$O^{(\mathrm{low})} = \sum_{|P| \leq k} \alpha_P P$$

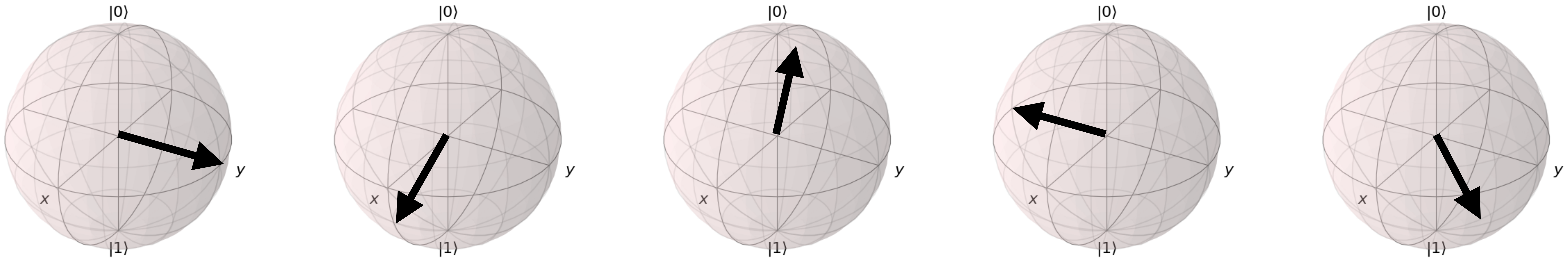
Classical inputs are perfectly distinguishable.

But quantum state inputs are not.

Classical Input:



Quantum Input:



Basic Idea for the ML model

Basic idea: Learn the low-weight observable $O^{(\mathrm{low})} = \sum_{|P| \leq k} \alpha_P P$ for a small k.

Lemma (Fourier transform): $\alpha_P = \mathbb{E}\left[\frac{3^{|P|}}{N} \sum_{\ell=1}^{N} y_\ell \langle \psi_\ell \,|\, P \,|\, \psi_\ell \rangle \right], \; \forall P \in \{I, X, Y, Z\}^{\otimes n}$

Classical Dataset

$$|\psi_\ell\rangle = \bigotimes_{i=1}^{n} |\psi_{\ell,i}\rangle \mapsto y_\ell, \quad \mathbb{E}[y_\ell] = \langle \psi_\ell | O | \psi_\ell \rangle$$

for $\ell = 1, \ldots, N$.

Basic Idea for the ML model

Basic idea: Learn the low-weight observable $O^{(\text{low})} = \sum_{|P| \leq k} \alpha_P P$ for a small k.

Lemma $, Z\}^{\otimes n}$

How large should the data size N be?

Classical Dataset

$$|\psi_\ell\rangle = \bigotimes_{i=1}^{n} |\psi_{\ell,i}\rangle \mapsto y_\ell, \quad \mathbb{E}[y_\ell] = \langle \psi_\ell | O | \psi_\ell \rangle$$

for $\ell = 1, \ldots, N$.

Basic Idea for the ML model

Basic idea: Learn the low-weight observable $O^{(\text{low})} = \sum_{|P| \leq k} \alpha_P P$ for a small k.

Lemma … $, Z\}^{\otimes n}$

We only need $N = \mathcal{O}(\log n)$!

Classical Dataset

$$|\psi_\ell\rangle = \bigotimes_{i=1}^{n} |\psi_{\ell,i}\rangle \mapsto y_\ell, \quad \mathbb{E}[y_\ell] = \langle \psi_\ell | O | \psi_\ell \rangle$$

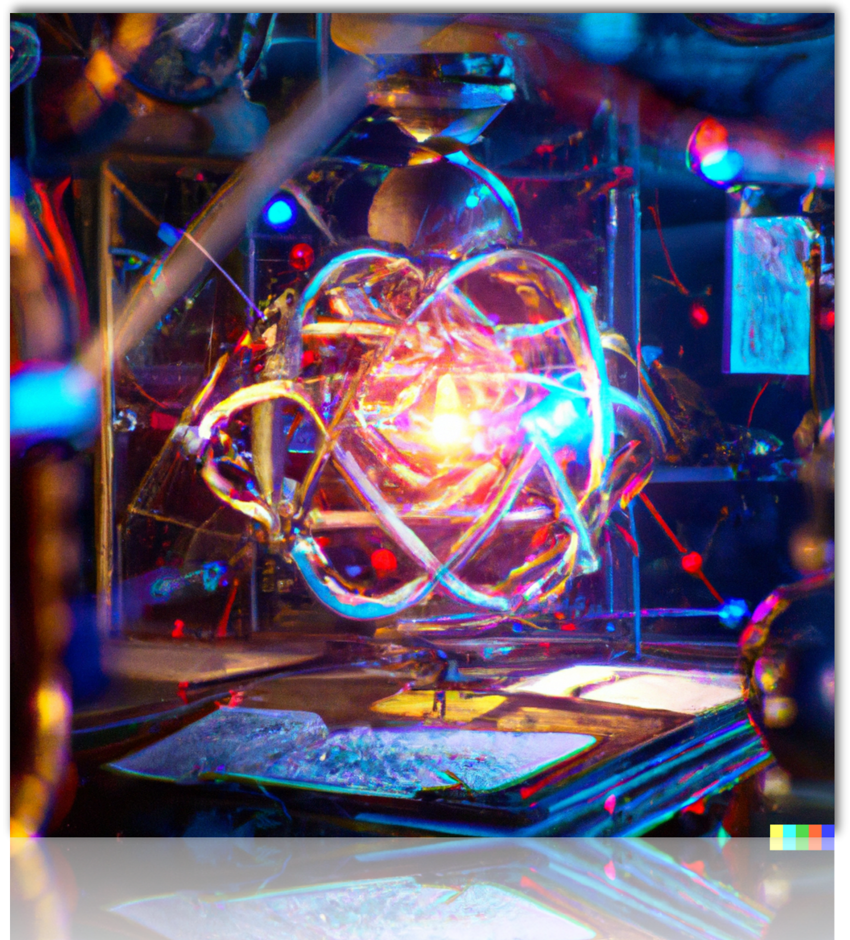
for $\ell = 1, \ldots, N$.

An interlude

Optimizing Quantum Hamiltonians

Presenter: Hsin-Yuan Huang (Robert)

Joint work with Sitan Chen and John Preskill



IQIM INSTITUTE FOR QUANTUM INFORMATION AND MATTER Caltech HARVARD UNIVERSITY aws

The Task

Given an n-qubit, k-local Hamiltonian $H = \sum_{|P| \leq k} \alpha_P P$.

Find a state $|\psi\rangle$ that maximizes or minimizes $\langle\psi|H|\psi\rangle$.

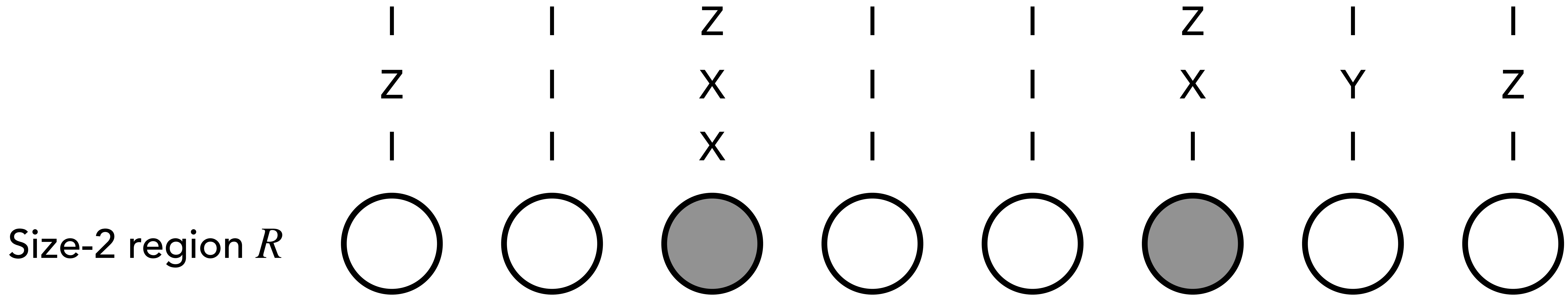
We want a guarantee on $\langle\psi|H|\psi\rangle$

based on the description of $H = \sum_{|P| \leq k} \alpha_P P$

Expansion property

Given an n-qubit, k-local Hamiltonian $H = \sum_{|P| \leq k} \alpha_P P$.

H has an expansion coefficient c_e and dimension d_e if for every size-d_e region R, the number of P with $\alpha_P \neq 0$, $\mathrm{dom}(P) \subseteq R, R \subseteq \mathrm{dom}(P)$ is at most c_e.



Expansion property

Given an n-qubit, k-local Hamiltonian $H = \sum_{|P| \leq k} \alpha_P P$.

H has an expansion coefficient c_e and dimension d_e if for every size-d_e region R, the number of P with $\alpha_P \neq 0, \mathrm{dom}(P) \subseteq R, R \subseteq \mathrm{dom}(P)$ is at most c_e.

Example 1

Geometrically-local Hamiltonian

$c_e = \mathcal{O}(1), d_e = 1$

Example 2

General k-local Hamiltonian

$c_e = 4^k, d_e = k$

Example 3

Degree-d 2-body Hamiltonian

$c_e = 16d, d_e = 1$

Theorem

Given an n-qubit, k-local Hamiltonian $H = \sum_{|P| \leq k} \alpha_P P$.

If H has an expansion coefficient c_e and dimension d_e, then for $r = 2d_e/(d_e + 1) \in [1,2)$,

we have an algorithm that either finds a maximizing product state $|\psi\rangle$,

$$\langle \psi | H | \psi \rangle \geq \mathbb{E}_{|\phi\rangle:\mathrm{Haar}} \langle \phi | H | \phi \rangle + \frac{1}{c_e^{1/2d_e} 2^{\Theta(k \log k)}} \left(\sum_{P \neq I} |\alpha_P|^r \right)^{1/r},$$

or finds a minimizing product state $|\psi\rangle$ with a similar guarantee ($+ \to -$, $\geq \to \leq$).

Theorem

Improved over existing results

Given an n-qubit, k-local Hamiltonian $H = \sum_{|P| \leq k} \alpha_P P$.

If H has an expansion coefficient c_e and dimension d_e, then for $r = 2d_e/(d_e + 1) \in [1,2)$, we have an algorithm that either finds a maximizing product state $|\psi\rangle$,

$$\langle \psi | H | \psi \rangle \geq \mathbb{E}_{|\phi\rangle:\mathrm{Haar}} \langle \phi | H | \phi \rangle + \frac{1}{c_e^{1/2d_e} 2^{\Theta(k \log k)}} \left(\sum_{P \neq I} |\alpha_P|^r \right)^{1/r},$$

or finds a minimizing product state $|\psi\rangle$ with a similar guarantee ($+ \to -$, $\geq \to \leq$).

The Algorithm

Given an n-qubit, k-local Hamiltonian $H = \sum_{|P| \leq k} \alpha_P P$.

Find a product state $|\psi\rangle$ that approximately optimizes $\langle\psi|H|\psi\rangle$.

The Algorithm

Select a slice with the largest value of α_P

Given an n-qubit, k-local Hamiltonian $H = \sum_{|P|=k} \alpha_P P$.

Find a product state $|\psi\rangle$ that approximately optimizes $\langle\psi|H|\psi\rangle$.

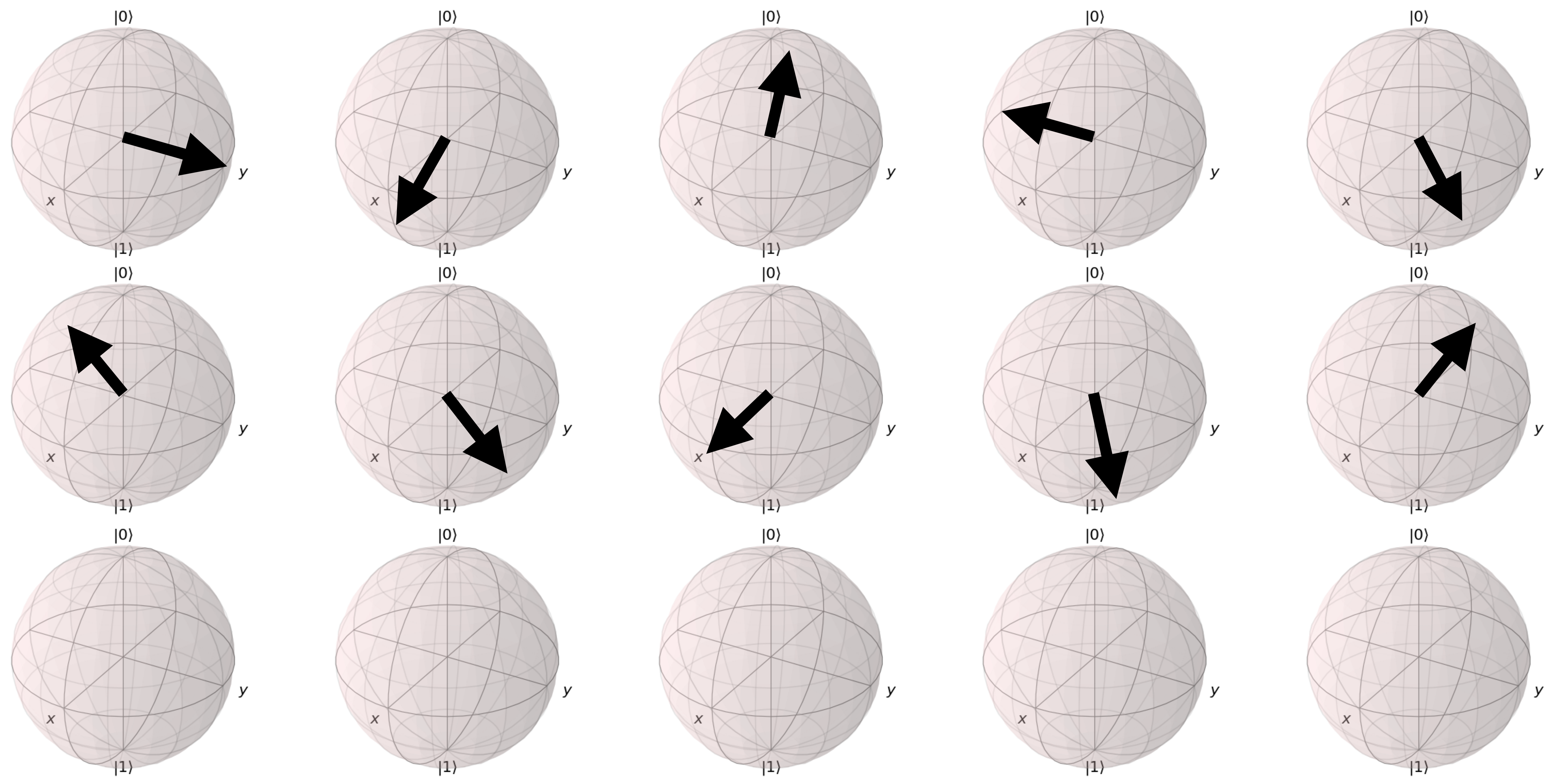
The Algorithm

Given an n-qubit, k-local Hamiltonian $H = \sum_{|P|=k} \alpha_P P$.

Replica 1

Replica 2

Replica 3



Random product states

The Algorithm

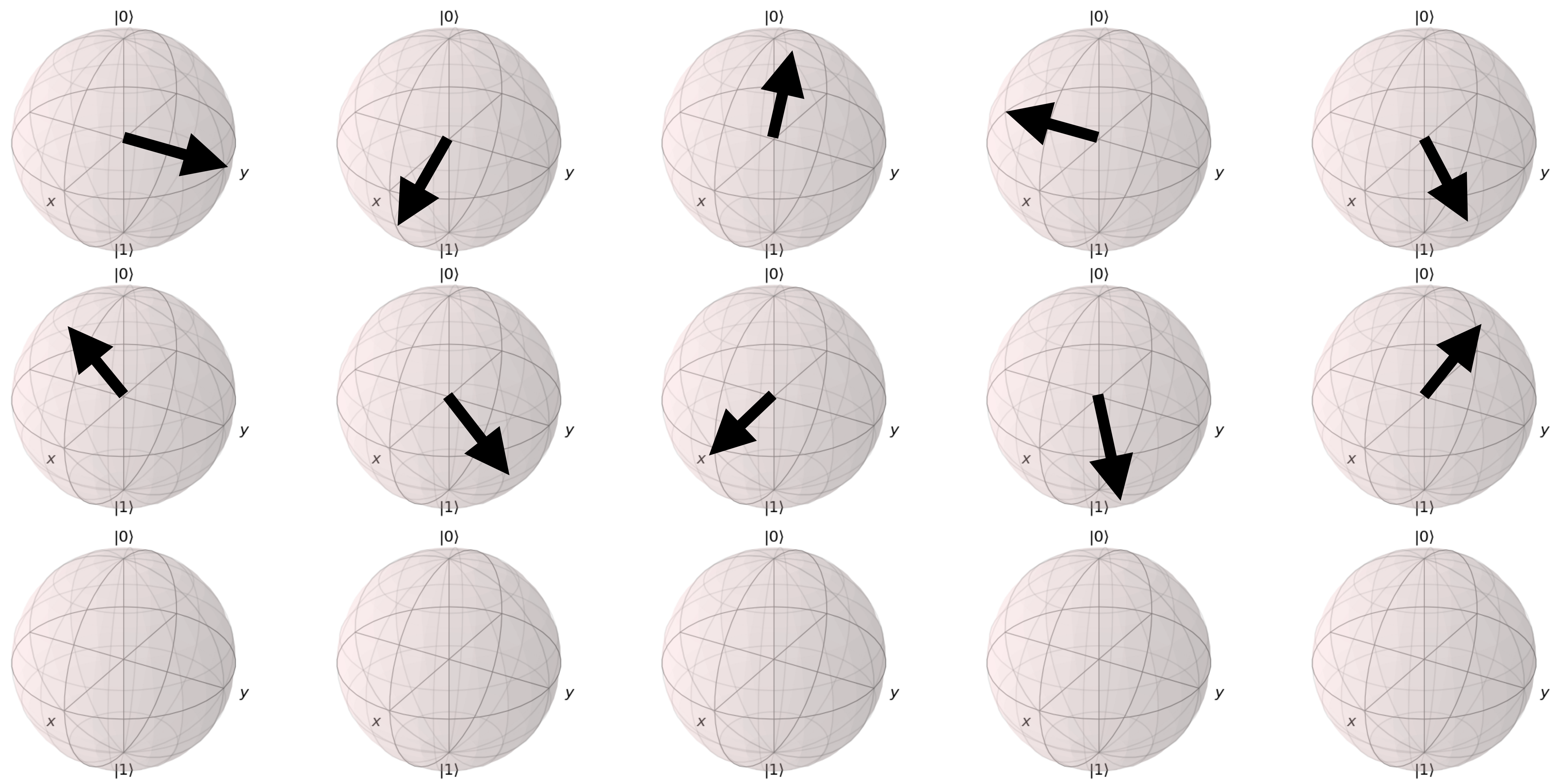
Lift n-qubit H to nk qubits

Given an n-qubit, k-local Hamiltonian $H = \sum_{|P|=k} \alpha_P P$.

Replica 1

Replica 2

Replica 3



$$\mathrm{pol}(H) = \sum_{|P|=k} \alpha_P \mathrm{pol}(P) \in \mathbb{C}^{2^{nk} \times 2^{nk}}$$

The Algorithm

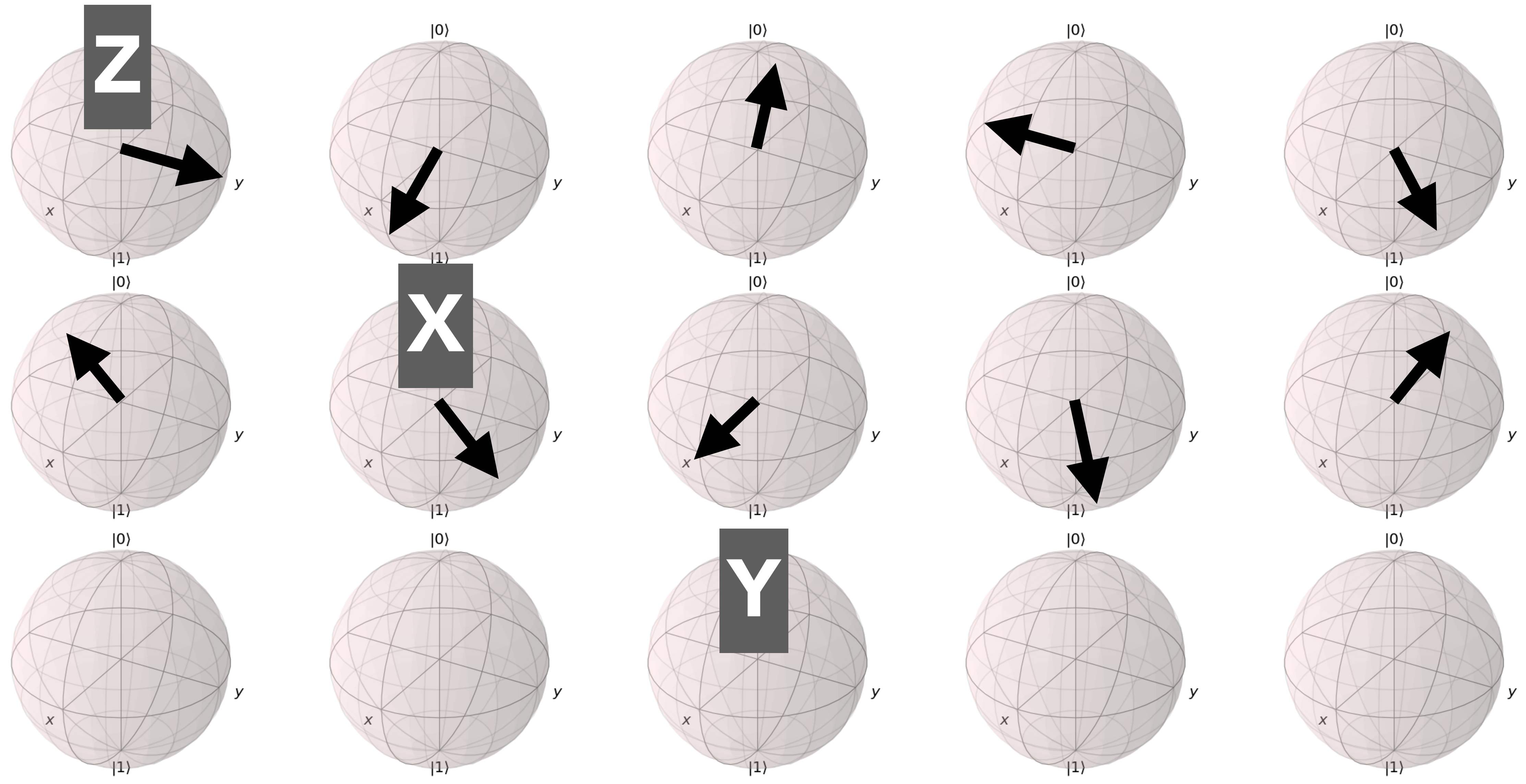
Lift n-qubit H to nk qubits

Given an n-qubit, k-local Hamiltonian $H = \sum_{|P|=k} \alpha_P P$.

Replica 1

Replica 2

Replica 3



$$\mathrm{pol}(H) = \sum_{|P|=k} \alpha_P \mathrm{pol}(P) \in \mathbb{C}^{2^{nk} \times 2^{nk}}$$

P = Z X Y I I

The Algorithm

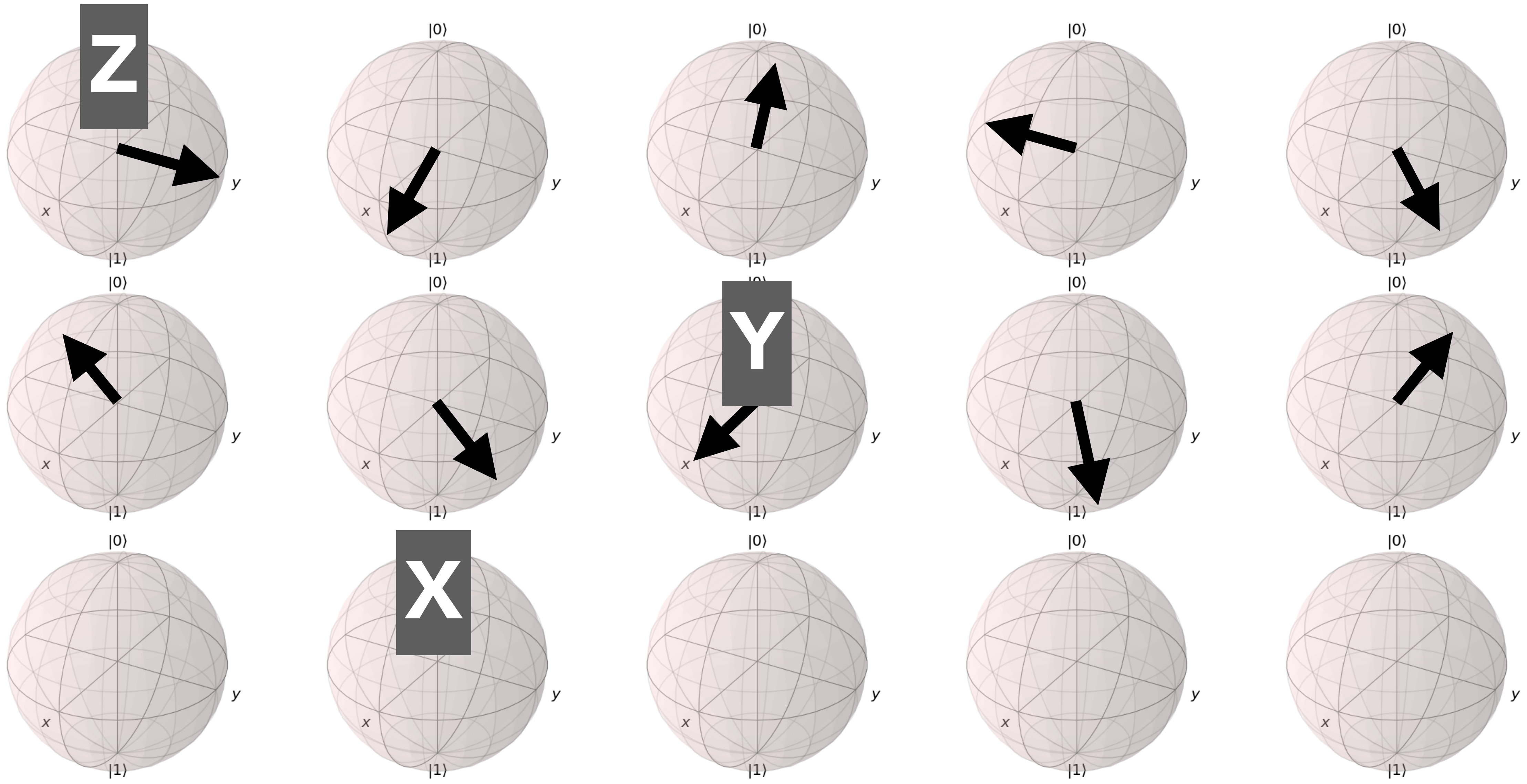
Lift n-qubit H to nk qubits

Given an n-qubit, k-local Hamiltonian $H = \sum_{|P|=k} \alpha_P P$.

Replica 1

Replica 2

Replica 3



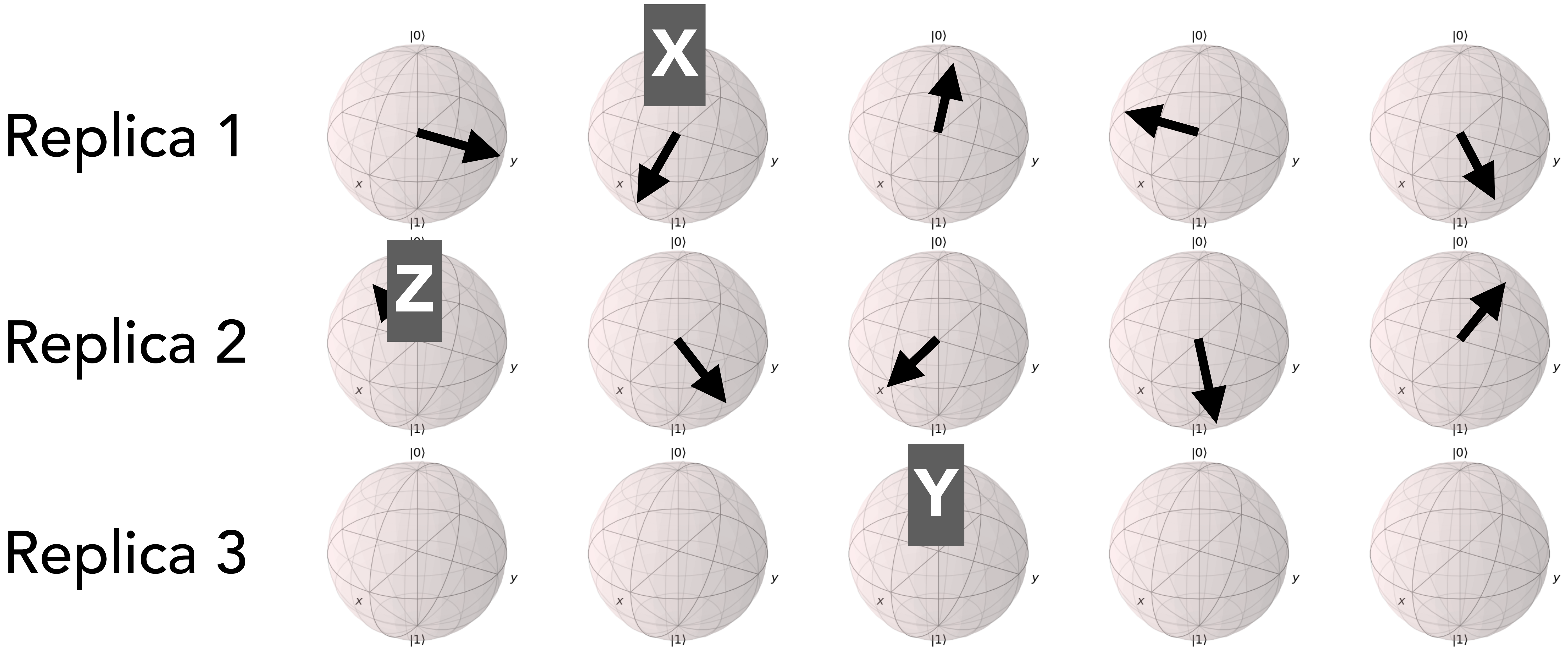
$$\mathrm{pol}(H) = \sum_{|P|=k} \alpha_P \mathrm{pol}(P) \in \mathbb{C}^{2^{nk} \times 2^{nk}}$$

P = Z X Y I I

The Algorithm

Lift n-qubit H to nk qubits

Given an n-qubit, k-local Hamiltonian $H = \sum_{|P|=k} \alpha_P P$.



$$\mathrm{pol}(H) = \sum_{|P|=k} \alpha_P \mathrm{pol}(P) \in \mathbb{C}^{2^{nk} \times 2^{nk}}$$

P = Z X Y I I

The Algorithm

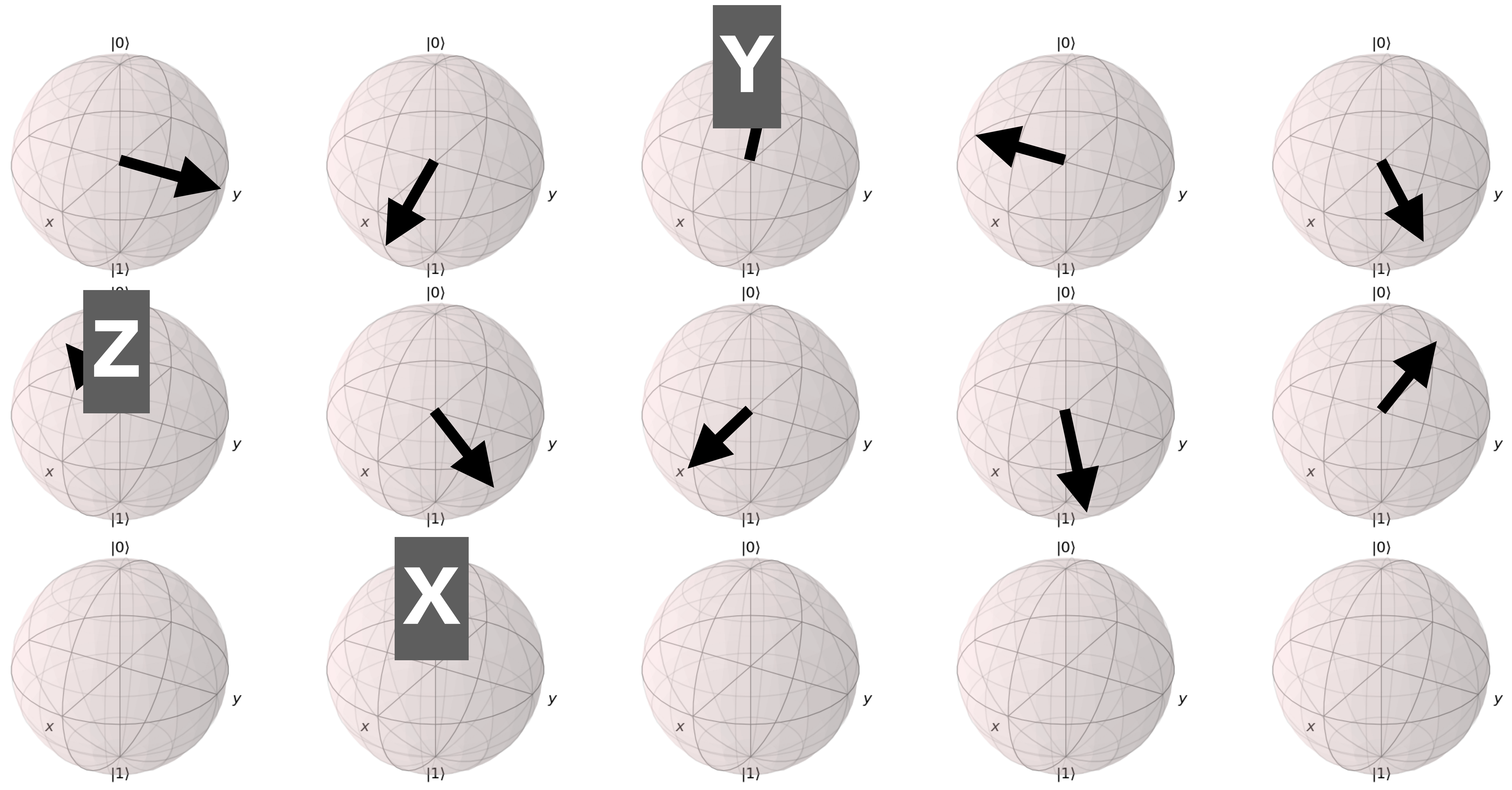
Lift n-qubit H to nk qubits

Given an n-qubit, k-local Hamiltonian $H = \sum_{|P|=k} \alpha_P P$.

Replica 1

Replica 2

Replica 3



$$\mathrm{pol}(H) = \sum_{|P|=k} \alpha_P \mathrm{pol}(P) \in \mathbb{C}^{2^{nk} \times 2^{nk}}$$

P = Z X Y I I

The Algorithm

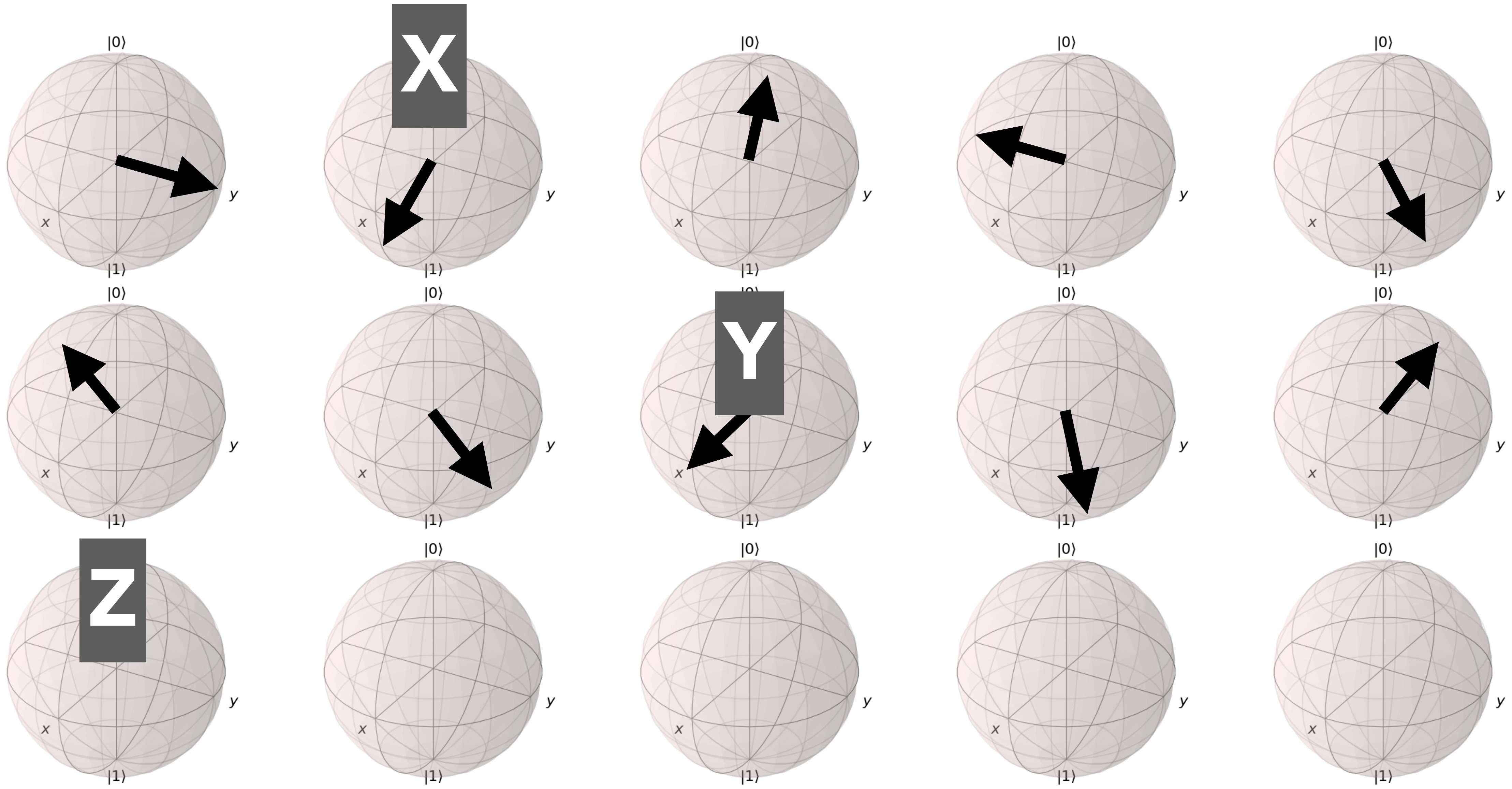
Lift n-qubit H to nk qubits

Given an n-qubit, k-local Hamiltonian $H = \sum_{|P|=k} \alpha_P P$.

Replica 1

Replica 2

Replica 3



$$\mathrm{pol}(H) = \sum_{|P|=k} \alpha_P \mathrm{pol}(P) \in \mathbb{C}^{2^{nk} \times 2^{nk}}$$

P = Z X Y I I

The Algorithm

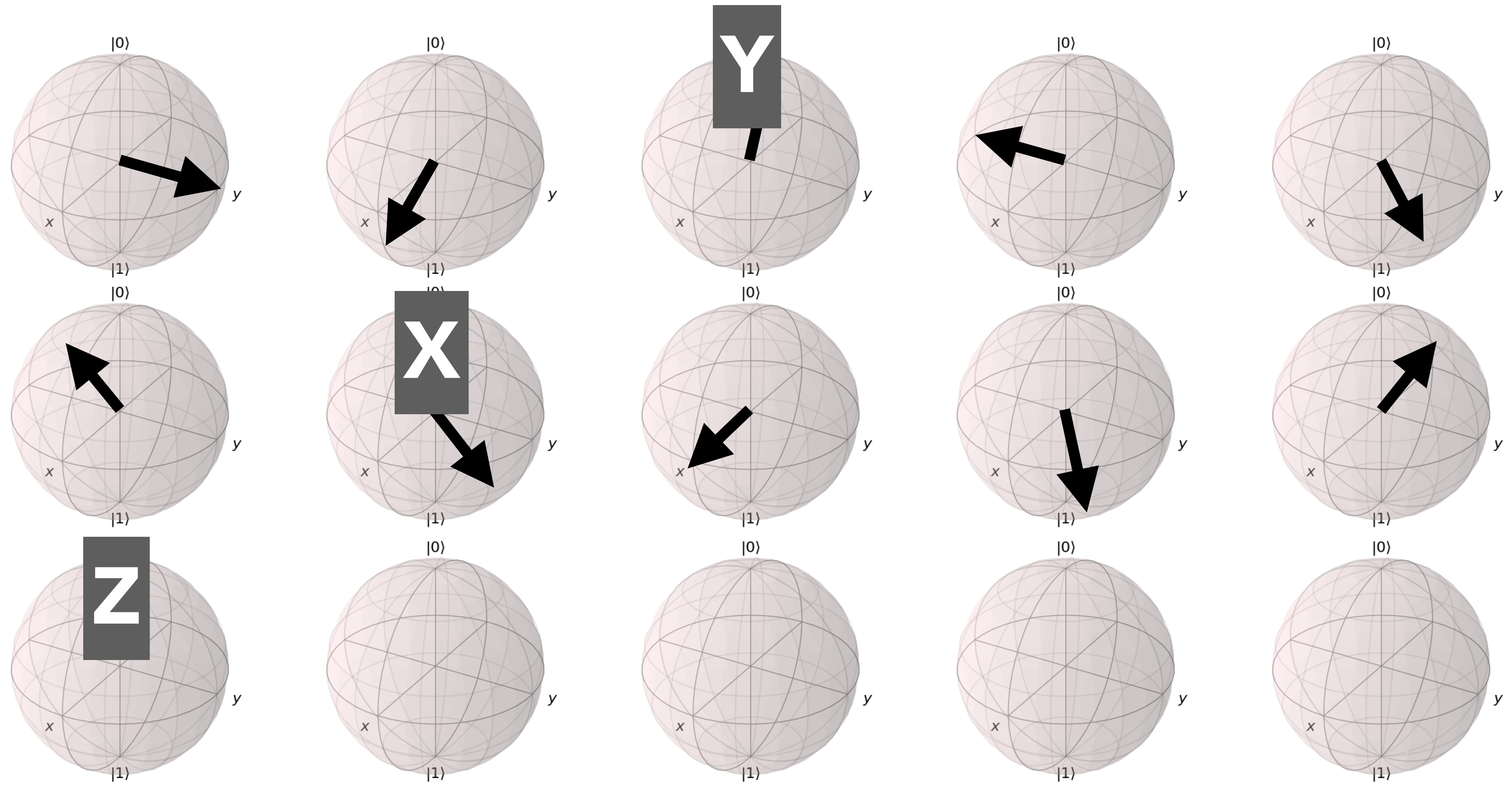
Lift n-qubit H to nk qubits

Given an n-qubit, k-local Hamiltonian $H = \sum_{|P|=k} \alpha_P P$.

Replica 1

Replica 2

Replica 3



$$\mathrm{pol}(H) = \sum_{|P|=k} \alpha_P \mathrm{pol}(P) \in \mathbb{C}^{2^{nk} \times 2^{nk}}$$

P = Z X Y I I

The Algorithm

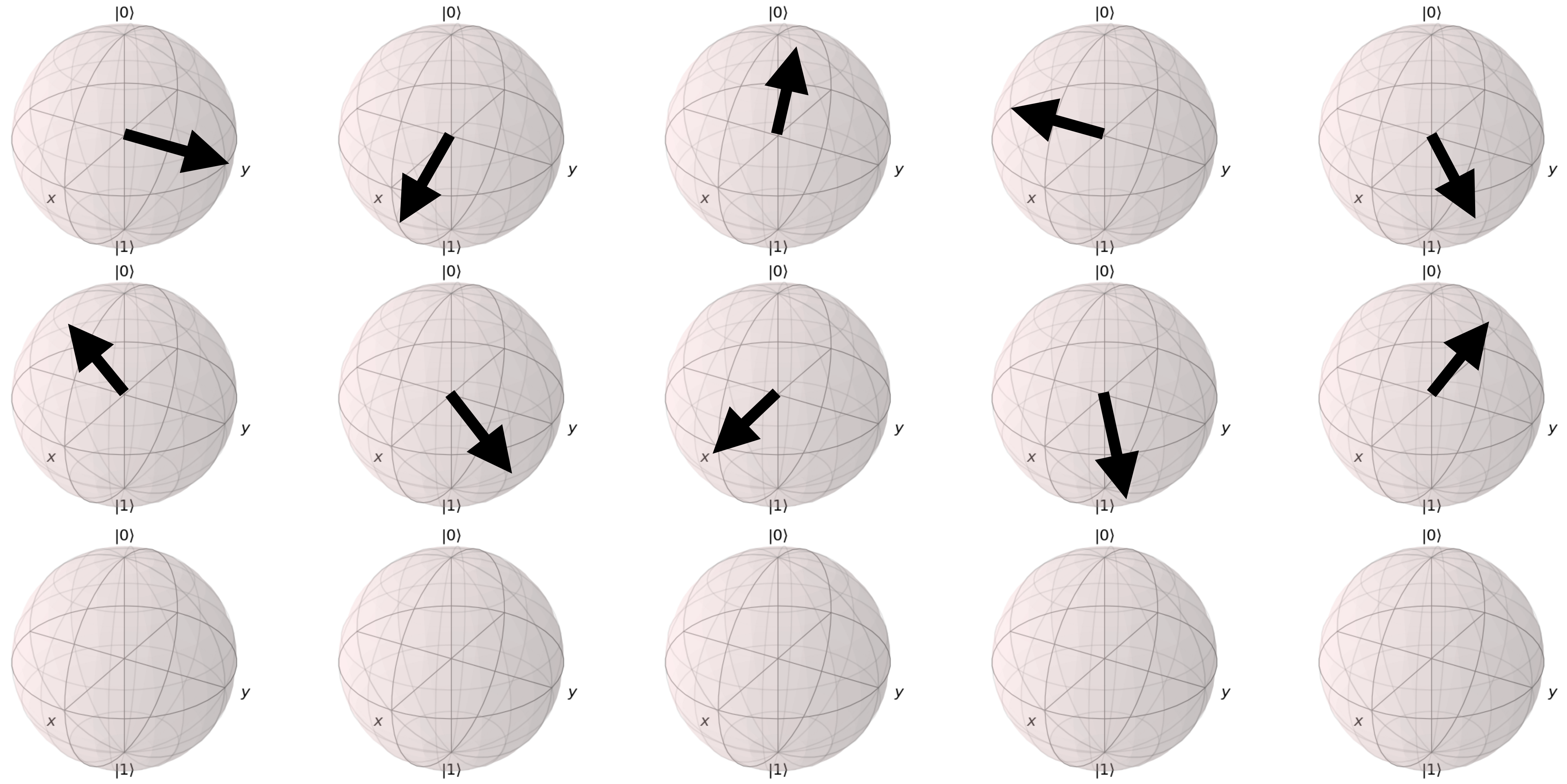
Lift n-qubit H to nk qubits

Given an n-qubit, k-local Hamiltonian $H = \sum_{|P|=k} \alpha_P P$.

Replica 1

Replica 2

Replica 3



$$\mathrm{pol}(H) = \sum_{|P|=k} \alpha_P \mathrm{pol}(P) \in \mathbb{C}^{2^{nk} \times 2^{nk}}$$

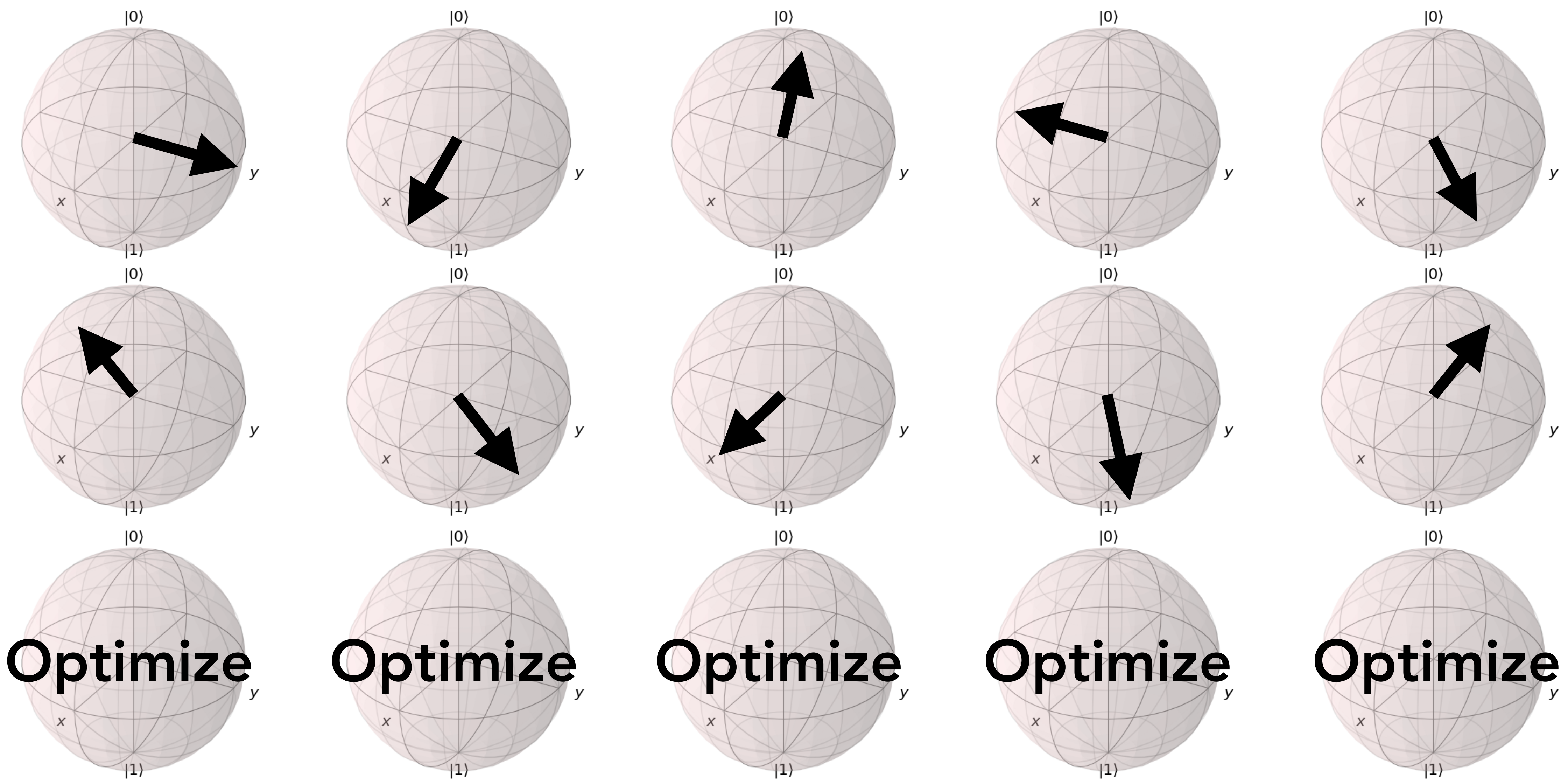
The Algorithm

Given an n-qubit, k-local Hamiltonian $H = \sum_{|P|=k} \alpha_P P$.

Replica 1

Replica 2

Replica 3



$$\mathrm{pol}(H) = \sum_{|P|=k} \alpha_P \mathrm{pol}(P) \in \mathbb{C}^{2^{nk} \times 2^{nk}}$$

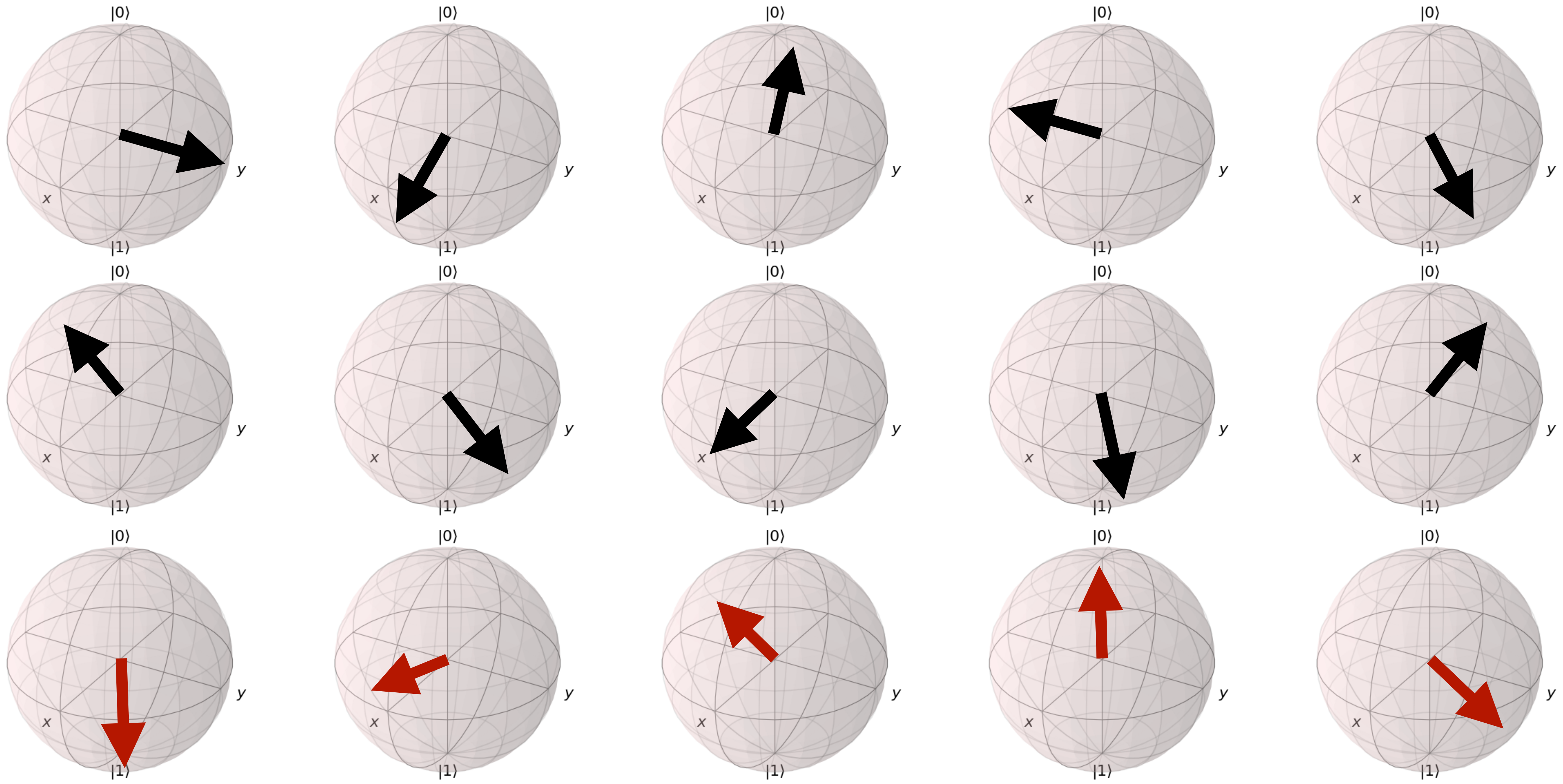
The Algorithm

Given an n-qubit, k-local Hamiltonian $H = \sum_{|P|=k} \alpha_P P$.

Replica 1

Replica 2

Replica 3

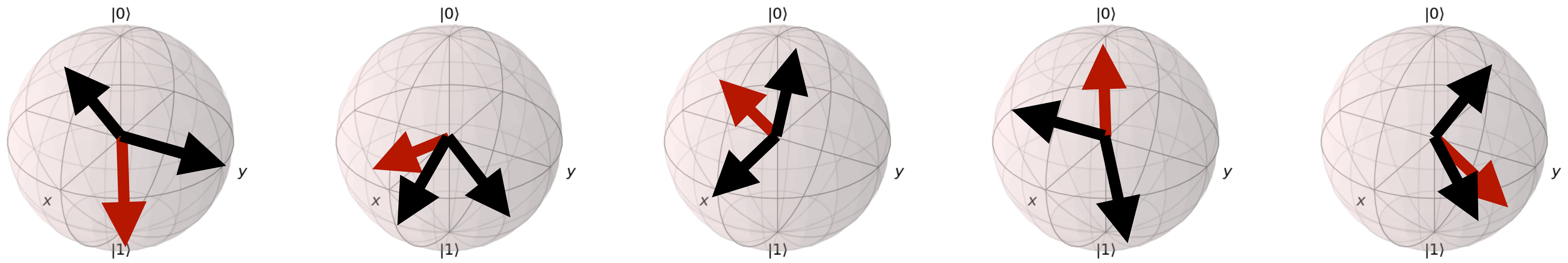


$$\mathrm{pol}(H) = \sum_{|P|=k} \alpha_P \mathrm{pol}(P) \in \mathbb{C}^{2^{nk} \times 2^{nk}}$$

The Algorithm

Given an n-qubit, k-local Hamiltonian $H = \sum_{|P|=k} \alpha_P P$.

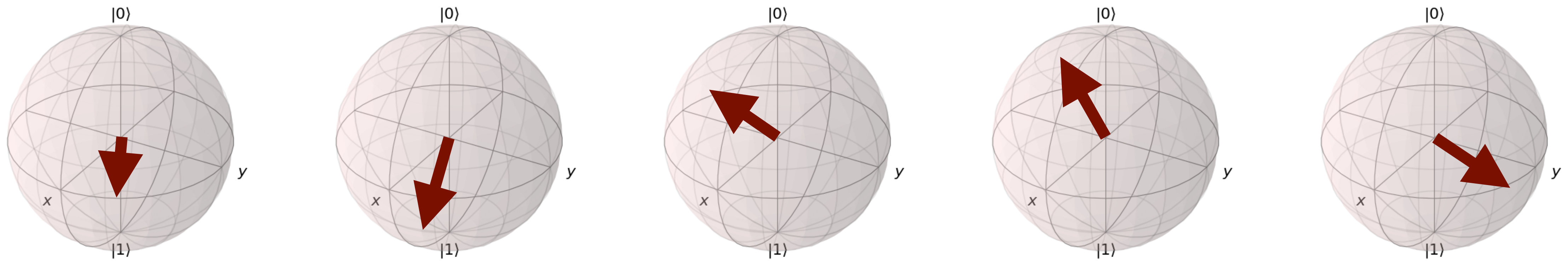
Replica 1
Replica 2
Replica 3



Combine the Bloch vectors using a weighted sum

The Algorithm

Given an n-qubit, k-local Hamiltonian $H = \sum_{|P|=k} \alpha_P P$.



Combine the Bloch vectors using a weighted sum

Theorem

Given an n-qubit, k-local Hamiltonian $H = \sum_{|P| \leq k} \alpha_P P$.

If H has an expansion coefficient c_e and dimension d_e, then for $r = 2d_e/(d_e + 1) \in [1,2)$,

we have an algorithm that either finds a maximizing product state $|\psi\rangle$,

$$\langle \psi | H | \psi \rangle \geq \mathbb{E}_{|\phi\rangle:\mathrm{Haar}} \langle \phi | H | \phi \rangle + \frac{1}{c_e^{1/2d_e} 2^{\Theta(k \log k)}} \left(\sum_{P \neq I} |\alpha_P|^r \right)^{1/r},$$

or finds a minimizing product state $|\psi\rangle$ with a similar guarantee ($+ \to -$, $\geq \to \leq$).

Another interlude

Generalized Quantum Bohnenblust-Hille Inequality

Presenter: Hsin-Yuan Huang (Robert)
Joint work with Sitan Chen and John Preskill

INSTITUTE FOR QUANTUM INFORMATION AND MATTER

Theorem

Given an observable $O = \sum_{|P| \leq k} \alpha_P P$ with an expansion coefficient c_e and dimension d_e.

$$\|O\|_\infty \geq \frac{1}{c_e^{1/2d_e} 2^{\Theta(k \log k)}} \left(\sum_P |\alpha_P|^r \right)^{1/r} \quad \text{for} \quad r = \frac{2d_e}{d_e + 1} \in [1,2).$$

Proof ideas:

(1) Use the guarantee from the algorithm for optimizing quantum Hamiltonians.

(2) Adapt by noting that $\|O\|_\infty \geq |\langle \psi | O | \psi \rangle|$, where $|\psi\rangle$ is the state found by the algo.

Theorem

Given an observable $O = \sum_{|P| \leq k} \alpha_P P$ with an expansion coefficient c_e and dimension d_e.

$$\|O\|_\infty \geq \frac{1}{c_e^{1/2d_e} 2^{\Theta(k \log k)}} \left(\sum_P |\alpha_P|^r \right)^{1/r} \quad \text{for} \quad r = \frac{2d_e}{d_e + 1} \in [1,2).$$

Example 1

A sum of geometrically-local terms

$c_e = \mathcal{O}(1)$, $d_e = 1$

$$\sum_P |\alpha_P| \leq \mathcal{O}\left(\|O\|_\infty\right)$$

Theorem

Given an observable $O = \sum_{|P| \leq k} \alpha_P P$ with an expansion coefficient c_e and dimension d_e.

$$\|O\|_\infty \geq \frac{1}{c_e^{1/2d_e} 2^{\Theta(k \log k)}} \left(\sum_P |\alpha_P|^r \right)^{1/r} \quad \text{for} \quad r = \frac{2d_e}{d_e + 1} \in [1,2).$$

Example 2

A sum of k-local terms

$c_e = 4^k, d_e = k$

$$\|\vec{\alpha}\|_{\frac{2k}{k+1}} \leq 2^{\mathcal{O}(k \log k)} \|O\|_\infty$$

A quantum analogue of
the Bohnenblust-Hille inequality

Back to the original talk

Learning to predict arbitrary quantum processes

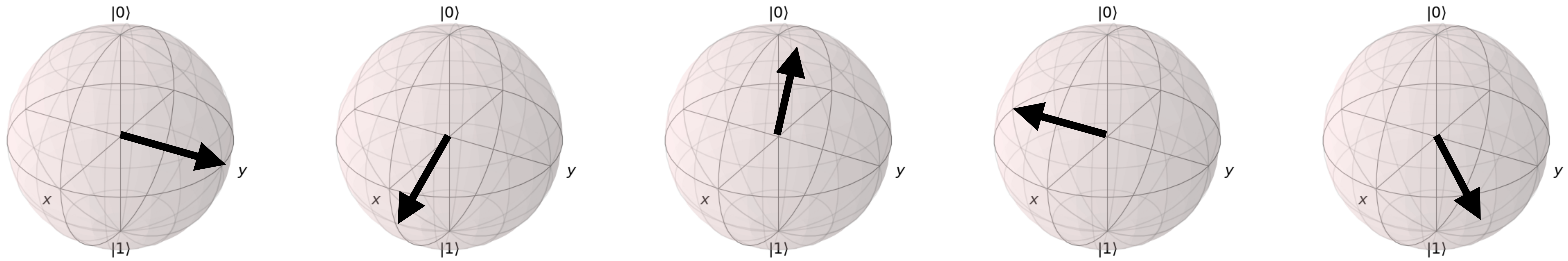
Presenter: Hsin-Yuan Huang (Robert)

Joint work with Sitan Chen and John Preskill

IQIM INSTITUTE FOR QUANTUM INFORMATION AND MATTER Caltech HARVARD UNIVERSITY aws

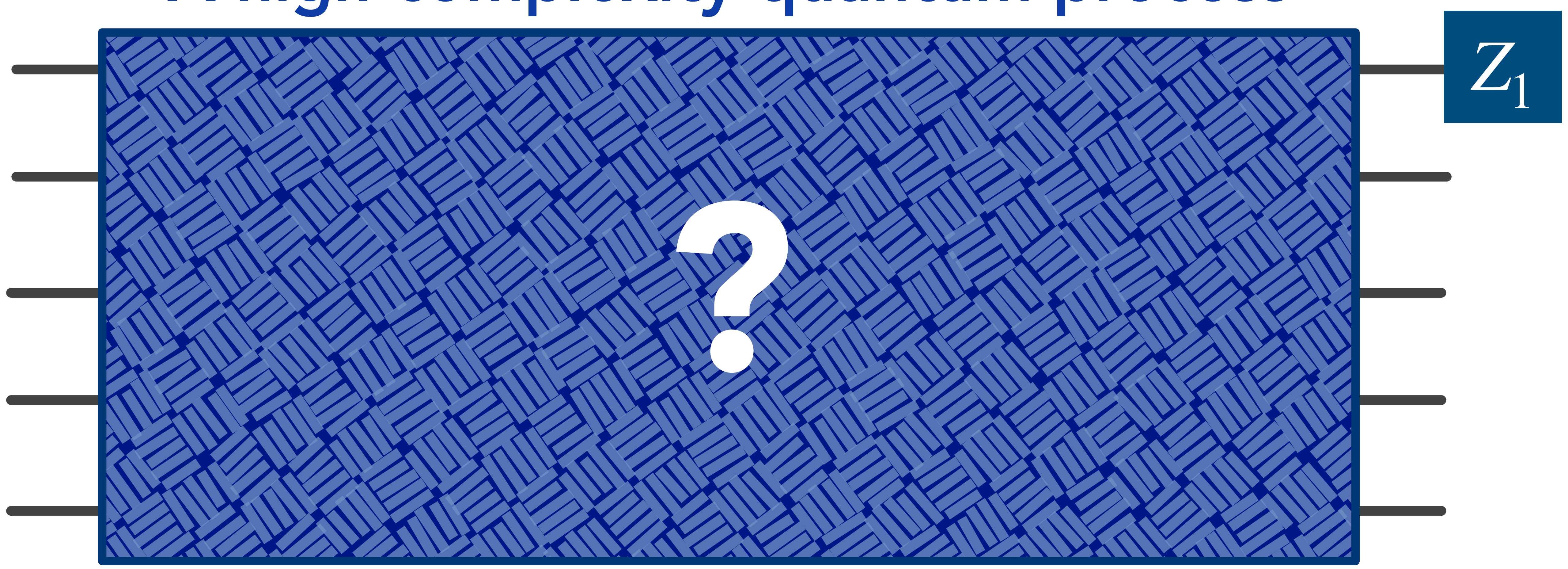
A Quantum Problem

Input:



$$\bigotimes_{i=1}^{n} |\psi_i\rangle \in (\mathbb{C}^2)^{\otimes n}$$

A high-complexity quantum process



Basic Idea for the ML model

Basic idea: Learn the low-weight observable $O^{(\mathrm{low})} = \sum_{|P| \leq k} \alpha_P P$ for a small k.

Lemma (Fourier transform): $\alpha_P = \mathbb{E}\left[\frac{3^{|P|}}{N} \sum_{\ell=1}^{N} y_\ell \langle \psi_\ell \,|\, P \,|\, \psi_\ell \rangle\right], \forall P \in \{I, X, Y, Z\}^{\otimes n}$

Classical Dataset

$$|\psi_\ell\rangle = \bigotimes_{i=1}^{n} |\psi_{\ell,i}\rangle \mapsto y_\ell, \quad \mathbb{E}[y_\ell] = \langle \psi_\ell | O | \psi_\ell \rangle$$

for $\ell = 1, \ldots, N$.

Basic Idea for the ML model

Basic idea: Learn the low-weight observable $O^{(\mathrm{low})} = \sum_{|P| \leq k} \alpha_P P$ for a small k.

Lemma $, Z\}^{\otimes n}$

How large should the data size N be?

Classical Dataset

$$|\psi_\ell\rangle = \bigotimes_{i=1}^{n} |\psi_{\ell,i}\rangle \mapsto y_\ell, \quad \mathbb{E}[y_\ell] = \langle \psi_\ell | O | \psi_\ell \rangle$$

for $\ell = 1, \ldots, N$.

Insight from Quantum BH inequality

Insight 1: Learn the low-weight observable $O^{(\text{low})} = \sum_{|P| \leq k} \alpha_P P$ for a small k.

Insight 2: The Pauli coef. in $O^{(\text{low})}$ is approximately sparse as $\| \vec{\alpha} \|_{\frac{2k}{k+1}} \leq 2^{\mathcal{O}(k \log k)} \| O^{(\text{low})} \|_{\infty}$.

This idea is also used in classical learning theory [AI22]

Classical Dataset

$$|\psi_\ell\rangle = \bigotimes_{i=1}^{n} |\psi_{\ell,i}\rangle \mapsto y_\ell, \quad \mathbb{E}[y_\ell] = \langle \psi_\ell | O | \psi_\ell \rangle$$

for $\ell = 1, \ldots, N$.

The ML algorithm

Insight 1: Learn the low-weight observable $O^{(\text{low})} = \sum_{|P| \leq k} \alpha_P P$ for a small k.

Insight 2: The Pauli coef. in $O^{(\text{low})}$ is approximately sparse as $\|\vec{\alpha}\|_{\frac{2k}{k+1}} \leq 2^{\mathcal{O}(k \log k)} \|O^{(\text{low})}\|_\infty$.

Classical Dataset

$$|\psi_\ell\rangle = \bigotimes_{i=1}^{n} |\psi_{\ell,i}\rangle \mapsto y_\ell, \quad \mathbb{E}[y_\ell] = \langle \psi_\ell | O | \psi_\ell \rangle$$

for $\ell = 1, \ldots, N$.

For all $|P| \leq k$,

set $\hat{\alpha}_P \leftarrow \frac{3^{|P|}}{N} \sum_{\ell=1}^{N} y_\ell \langle \psi_\ell | P | \psi_\ell \rangle$.

If $\hat{\alpha}_P$ is small, set $\hat{\alpha}_P \leftarrow 0$.

The learned observable is $\hat{O}^{(\text{low})} = \sum_{|P| \leq k} \hat{\alpha}_P P$.

Guarantee for learning $O^{(\mathrm{low})}$

For any small constant ϵ, given a training set size $N = \mathcal{O}(\log n)$, the prediction error is

$$\mathbb{E}_{|\psi\rangle = \bigotimes_{i=1}^{n} |\psi_i\rangle} \left| \langle \psi | \hat{O}^{(\mathrm{low})} | \psi \rangle - \langle \psi | O^{(\mathrm{low})} | \psi \rangle \right|^2 < \epsilon \| O^{(\mathrm{low})} \|_\infty^2 .$$

Classical Dataset

$$|\psi_\ell\rangle = \bigotimes_{i=1}^{n} |\psi_{\ell,i}\rangle \mapsto y_\ell, \quad \mathbb{E}[y_\ell] = \langle \psi_\ell | O | \psi_\ell \rangle$$

for $\ell = 1, \ldots, N$.

For all $|P| \leq k$,

set $\hat{\alpha}_P \leftarrow \frac{3^{|P|}}{N} \sum_{\ell=1}^{N} y_\ell \langle \psi_\ell | P | \psi_\ell \rangle$.

If $\hat{\alpha}_P$ is small, set $\hat{\alpha}_P \leftarrow 0$.

The learned observable is $\hat{O}^{(\mathrm{low})} = \sum_{|P| \leq k} \hat{\alpha}_P P$.

Guarantee for learning O

For any small constant ϵ, ϵ', given a training set size $N = \mathcal{O}(\log n)$, the prediction error is

$$\mathbb{E}_{|\psi\rangle = \bigotimes_{i=1}^{n} |\psi_i\rangle} \left| \langle \psi | \hat{O}^{(\text{low})} | \psi \rangle - \langle \psi | O | \psi \rangle \right|^2 < \epsilon + \epsilon' \| O^{(\text{low})} \|_\infty^2 .$$

Classical Dataset

$$|\psi_\ell\rangle = \bigotimes_{i=1}^{n} |\psi_{\ell,i}\rangle \mapsto y_\ell, \quad \mathbb{E}[y_\ell] = \langle \psi_\ell | O | \psi_\ell \rangle$$

for $\ell = 1, \ldots, N$.

For all $|P| \leq k$,

set $\hat{\alpha}_P \leftarrow \frac{3^{|P|}}{N} \sum_{\ell=1}^{N} y_\ell \langle \psi_\ell | P | \psi_\ell \rangle$.

If $\hat{\alpha}_P$ is small, set $\hat{\alpha}_P \leftarrow 0$.

The learned observable is $\hat{O}^{(\text{low})} = \sum_{|P| \leq k} \hat{\alpha}_P P$.

Guarantee for learning O

For any ϵ, ϵ', given a training set size $N = \log(n)\, 2^{\tilde{\mathcal{O}}(\log(1/\epsilon)\log(1/\epsilon'))}$, the prediction error is

$$\mathbb{E}_{|\psi\rangle = \bigotimes_{i=1}^{n} |\psi_i\rangle} \left| \langle \psi | \hat{O}^{(\mathrm{low})} | \psi \rangle - \langle \psi | O | \psi \rangle \right|^2 < \epsilon + \epsilon' \| O^{(\mathrm{low})} \|_\infty^2.$$

Classical Dataset

$$|\psi_\ell\rangle = \bigotimes_{i=1}^{n} |\psi_{\ell,i}\rangle \mapsto y_\ell, \quad \mathbb{E}[y_\ell] = \langle \psi_\ell | O | \psi_\ell \rangle$$

for $\ell = 1, \ldots, N$.

For all $|P| \leq k$,

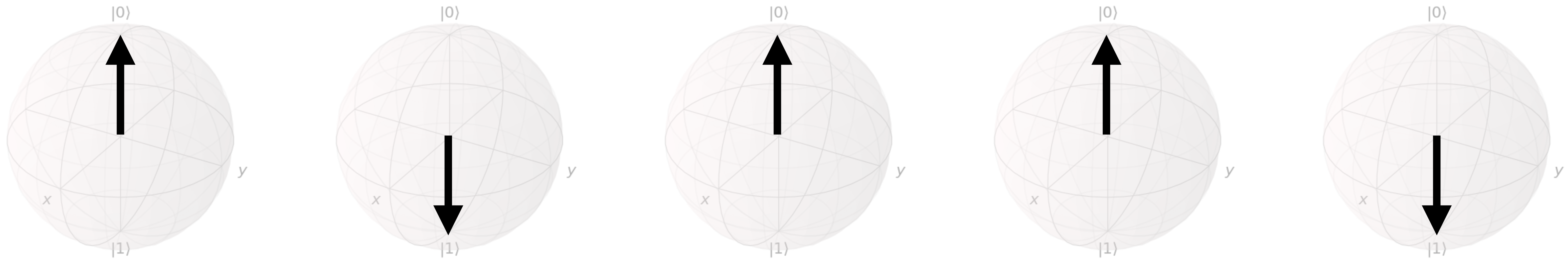
set $\hat{\alpha}_P \leftarrow \frac{3^{|P|}}{N} \sum_{\ell=1}^{N} y_\ell \langle \psi_\ell | P | \psi_\ell \rangle$.

If $\hat{\alpha}_P$ is small, set $\hat{\alpha}_P \leftarrow 0$.

The learned observable is $\hat{O}^{(\mathrm{low})} = \sum_{|P| \leq k} \hat{\alpha}_P P$.

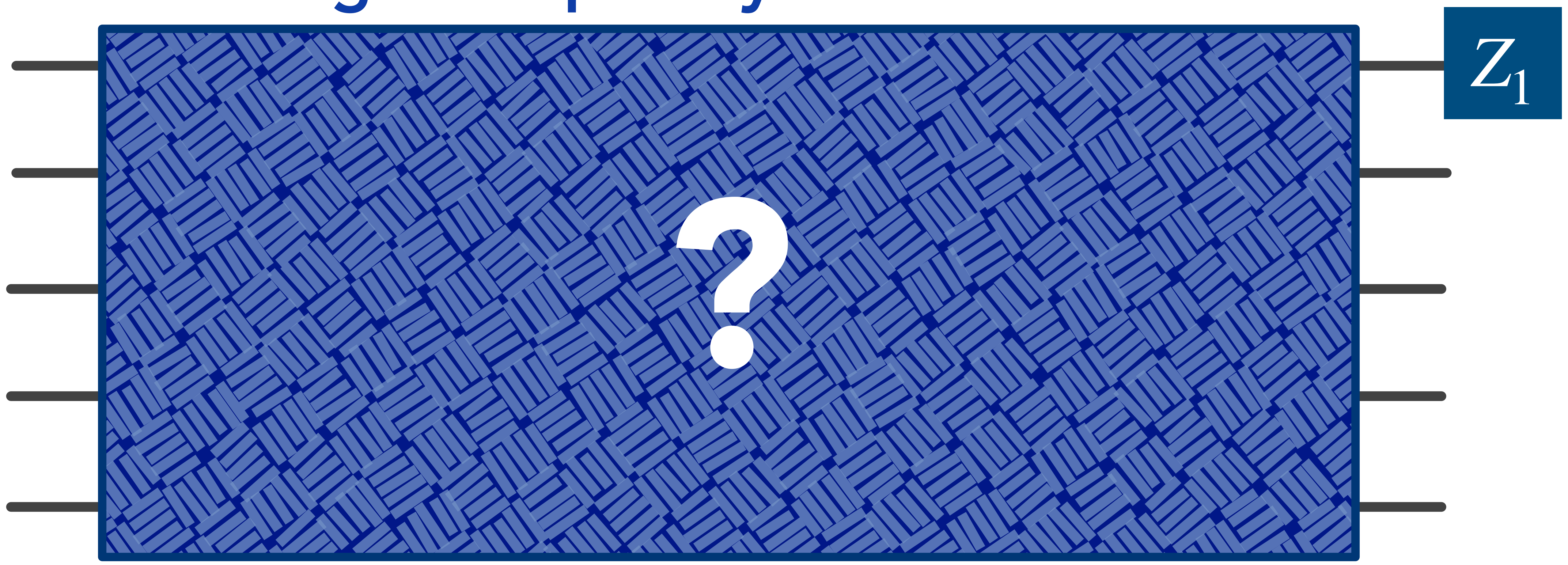
A Classical Problem

Input:



A high-complexity classical circuit

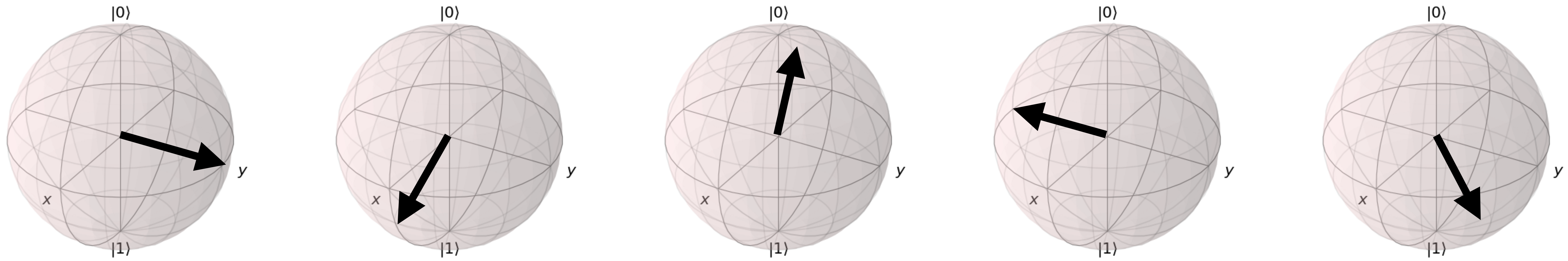
$x \in \{-1, 1\}^n$



Exponentially hard!

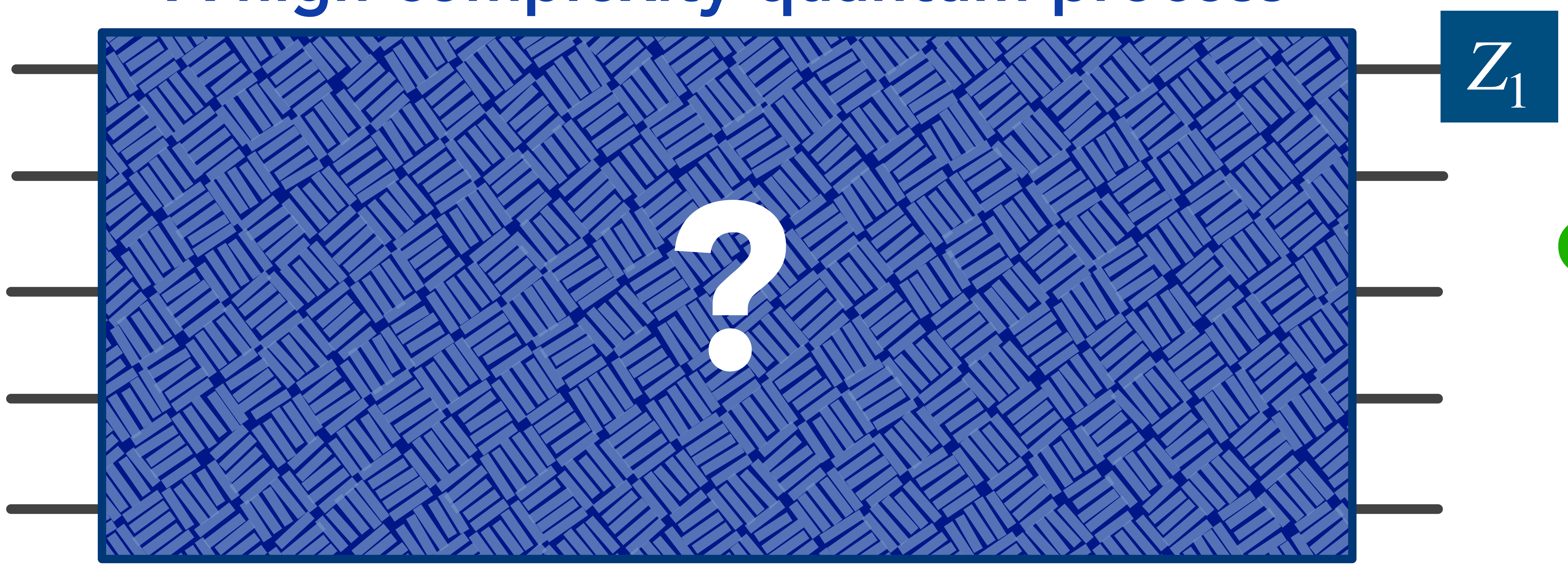
A Quantum Problem

Input:



$$\bigotimes_{i=1}^{n} |\psi_i\rangle \in (\mathbb{C}^2)^{\otimes n}$$

A high-complexity quantum process



Quasi-polynomially easy!

Overview

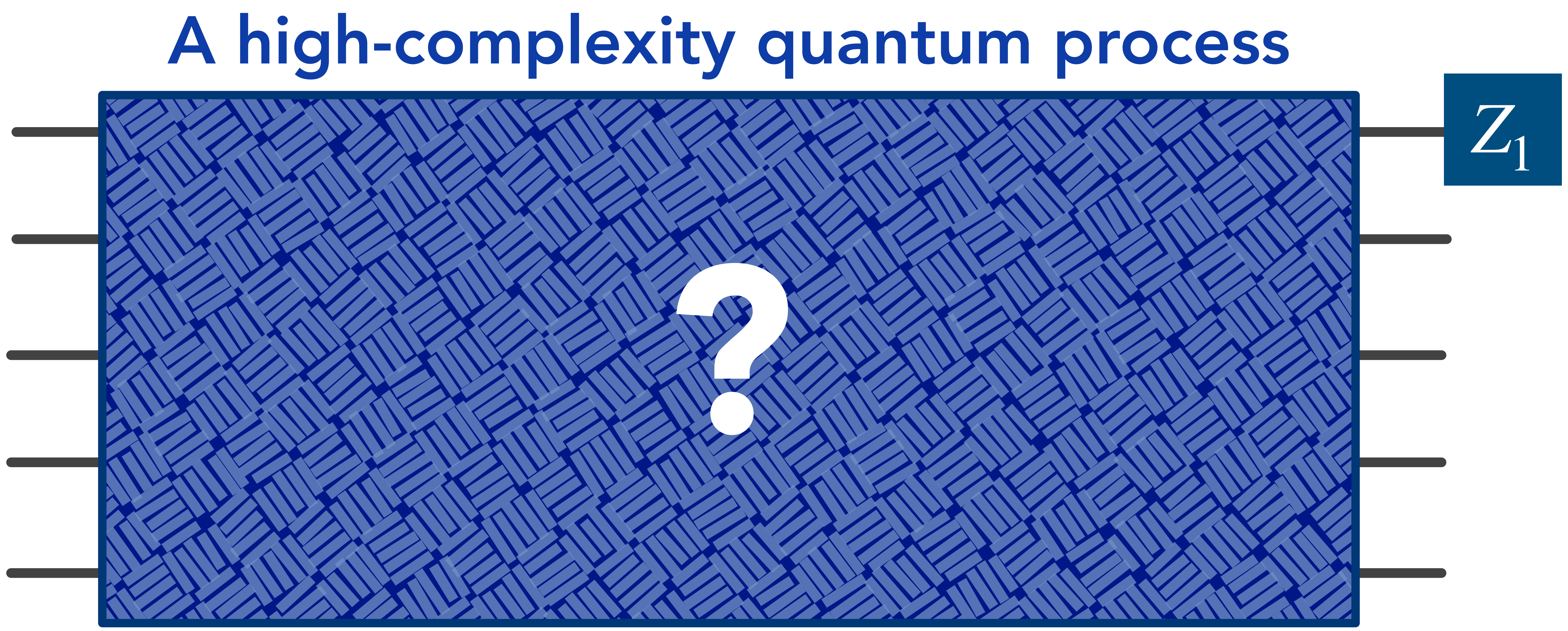
- A classical version of the quantum problem
- A restricted version of the quantum problem
- Generalization to the original quantum problem

Overview

- A classical version of the quantum problem
- A restricted version of the quantum problem
- Generalization to the original quantum problem

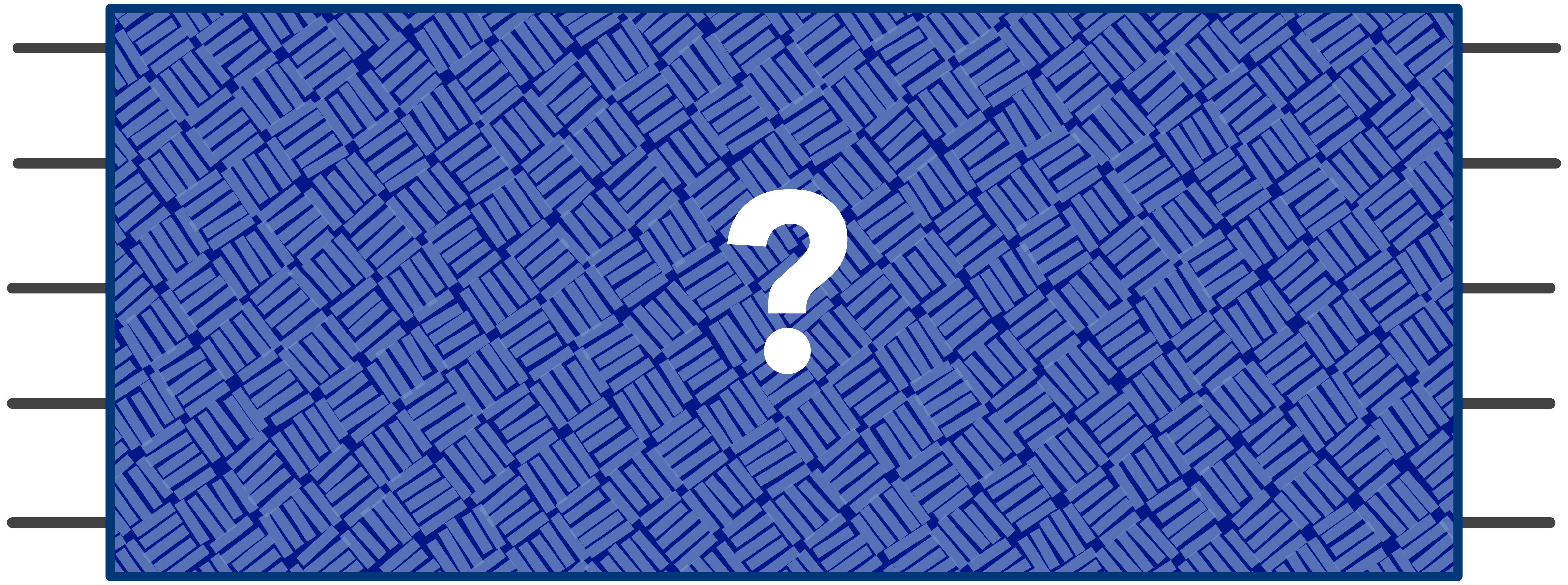
The Restricted Problem

$$\bigotimes_{i=1}^{n} |\psi_i\rangle \in (\mathbb{C}^2)^{\otimes n}$$



The Original Problem

A high-complexity quantum process



Entangled State

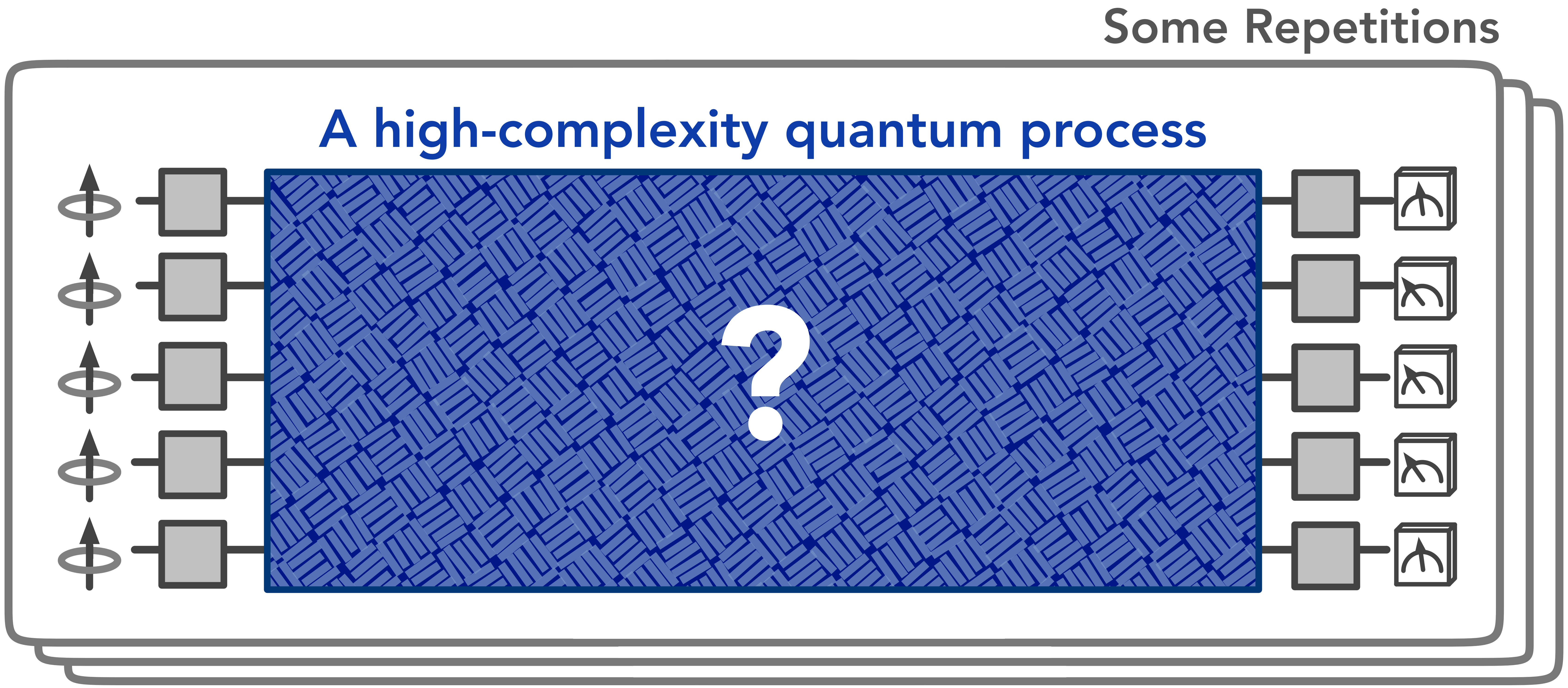
ρ

Observable

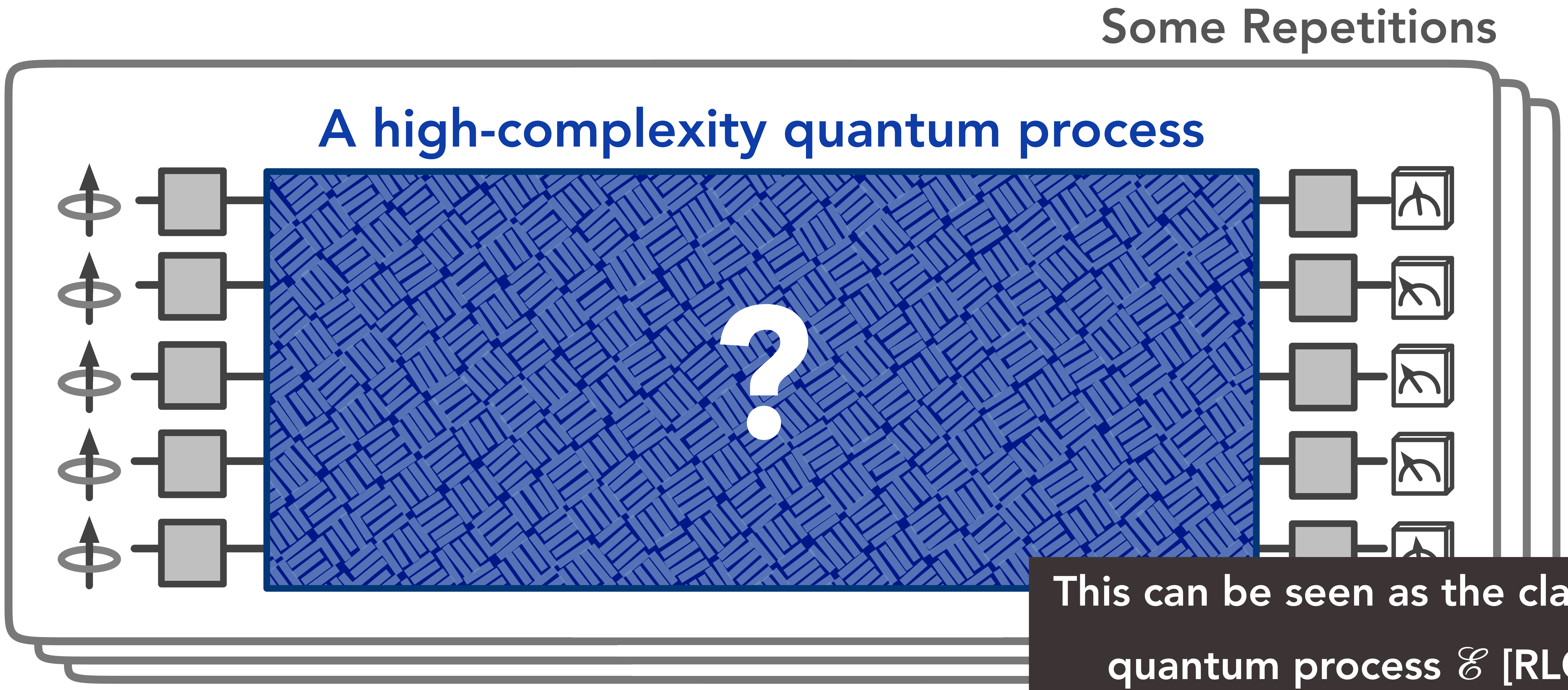
$$B = \sum_P \beta_P P$$

with $\|B\|_\infty \leq 1$

A Classical Dataset for Learning $\mathscr{E}$



A Classical Dataset for Learning $\mathscr{E}$



This can be seen as the classical shadow of quantum process $\mathscr{E}$ [RLC21, KTC+21]

A Classical Dataset for Learning $\mathscr{E}$

Classical Dataset

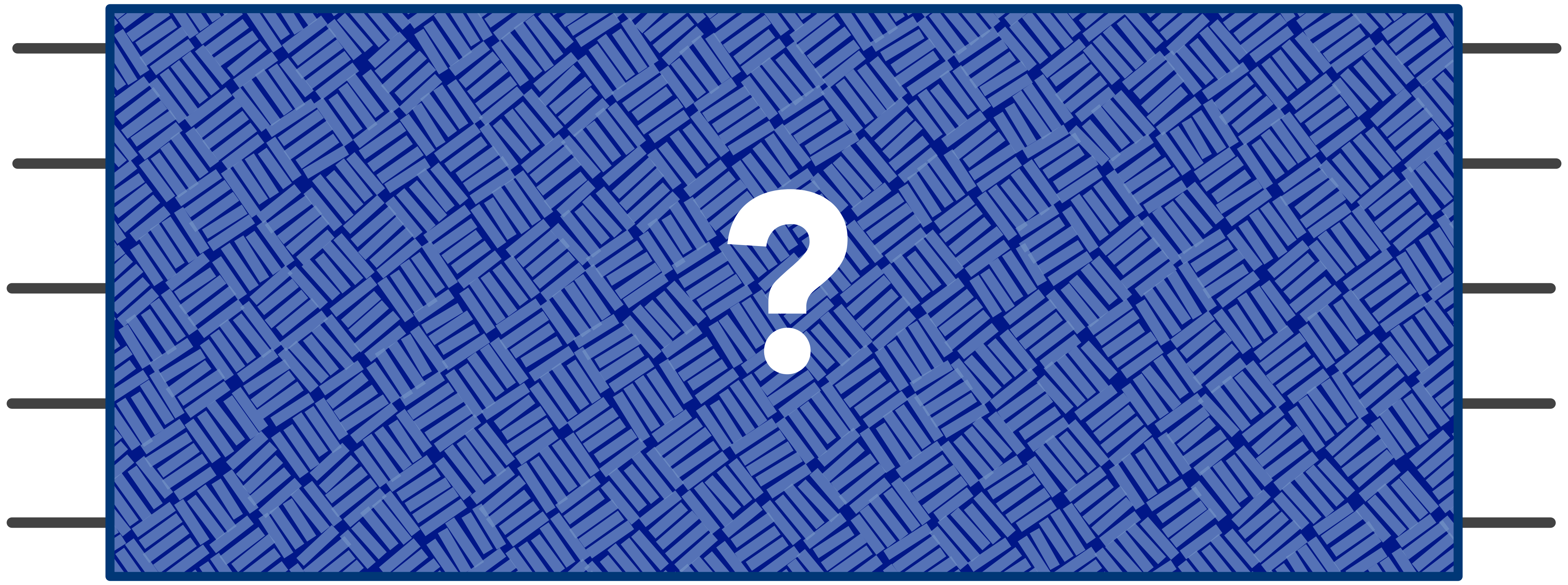
$$|\psi_\ell\rangle = \bigotimes_{i=1}^{n} |\psi_{\ell,i}\rangle \mapsto |\phi_\ell\rangle = \bigotimes_{i=1}^{n} |\phi_{\ell,i}\rangle$$

for $\ell = 1, \ldots, N$.

This can be seen as the classical shadow of quantum process $\mathscr{E}$ [RLC21, KTC+21]

How to make prediction?

A high-complexity quantum process



State

ρ

Observable

$$B = \sum_P \beta_P P$$

with $\|B\|_\infty \leq 1$

Classical Dataset

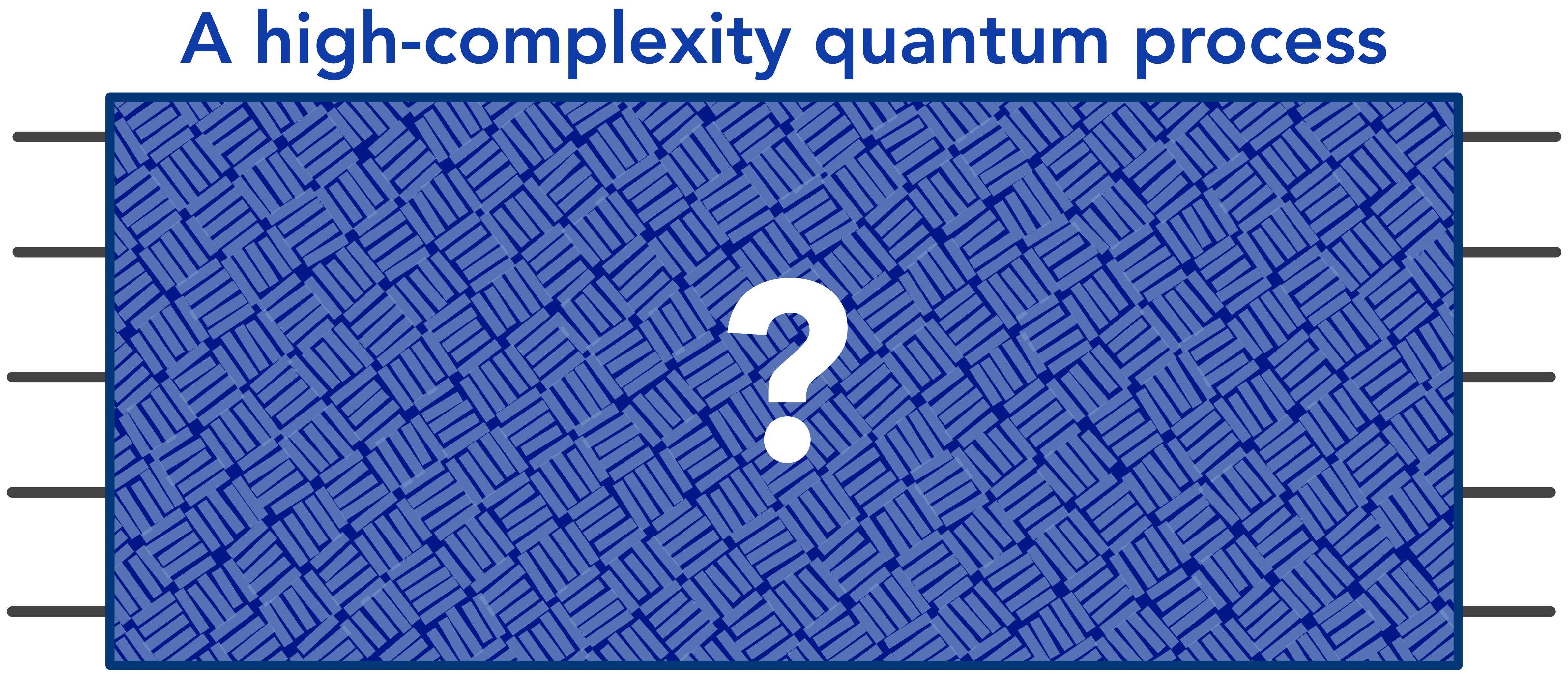
$$|\psi_\ell\rangle = \bigotimes_{i=1}^{n} |\psi_{\ell,i}\rangle \mapsto |\phi_\ell\rangle = \bigotimes_{i=1}^{n} |\phi_{\ell,i}\rangle$$

for $\ell = 1, \ldots, N.$

Construct a dataset with classical shadow

State

ρ



Observable

$$B = \sum_P \beta_P P$$

with $\|B\|_\infty \leq 1$

A New Classical Dataset

$$|\psi_\ell\rangle = \bigotimes_{i=1}^{n} |\psi_{\ell,i}\rangle \mapsto y_\ell = \mathrm{Tr}\left(B \bigotimes_{i=1}^{n} \left(3|\phi_{\ell,i}\rangle\langle\phi_{\ell,i}| - I\right) \right)$$

for $\ell = 1, \ldots, N.$

Properties [HKP20]:

$$\mathbb{E}[y_\ell] = \mathrm{Tr}(B\mathscr{E}(|\psi_\ell\rangle\langle\psi_\ell|))$$

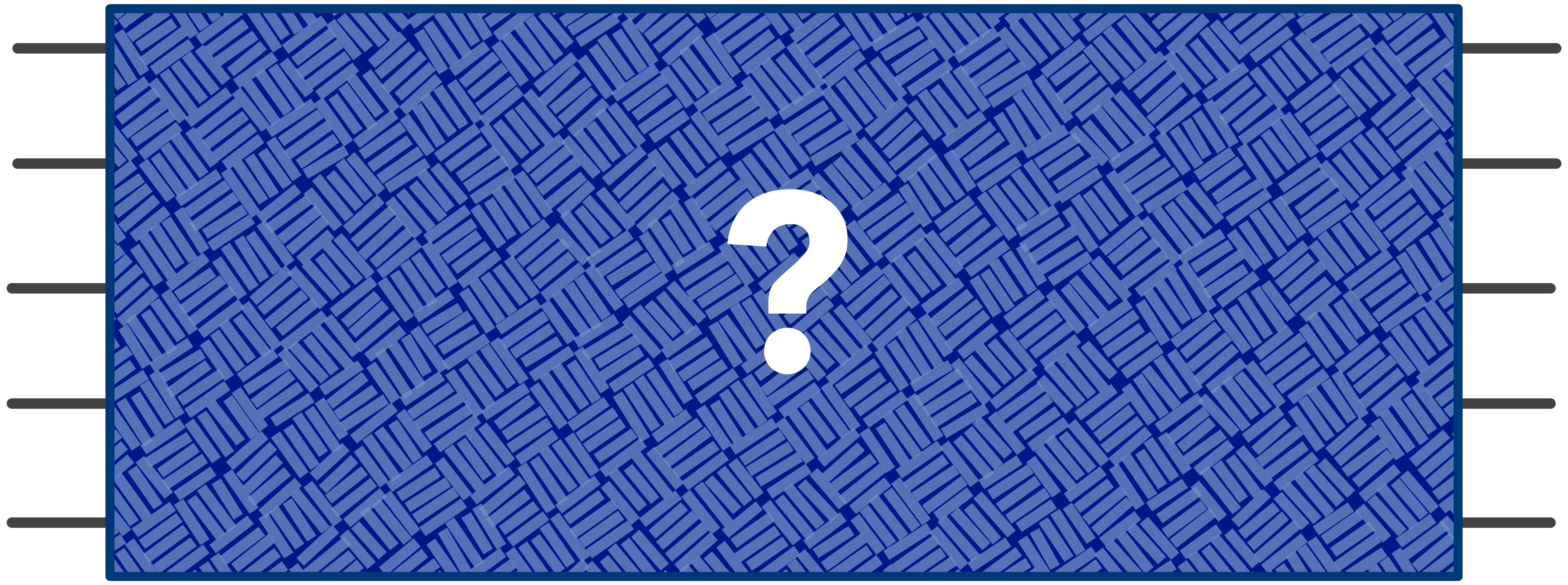
$$\mathrm{Var}[y_\ell] \leq \|B\|^2_{\mathrm{shadow}}$$

Construct a dataset with classical shadow

A high-complexity quantum process

State

ρ



Observable

$$B = \sum_P \beta_P P$$

with $\|B\|_\infty \leq 1$

A New Classical Dataset

$$|\psi_\ell\rangle = \bigotimes_{i=1}^{n} |\psi_{\ell,i}\rangle \mapsto y_\ell = \mathrm{Tr}\left(B \bigotimes_{i=1}^{n} \left(3|\phi_{\ell,i}\rangle\langle\phi_{\ell,i}| - I\right) \right)$$

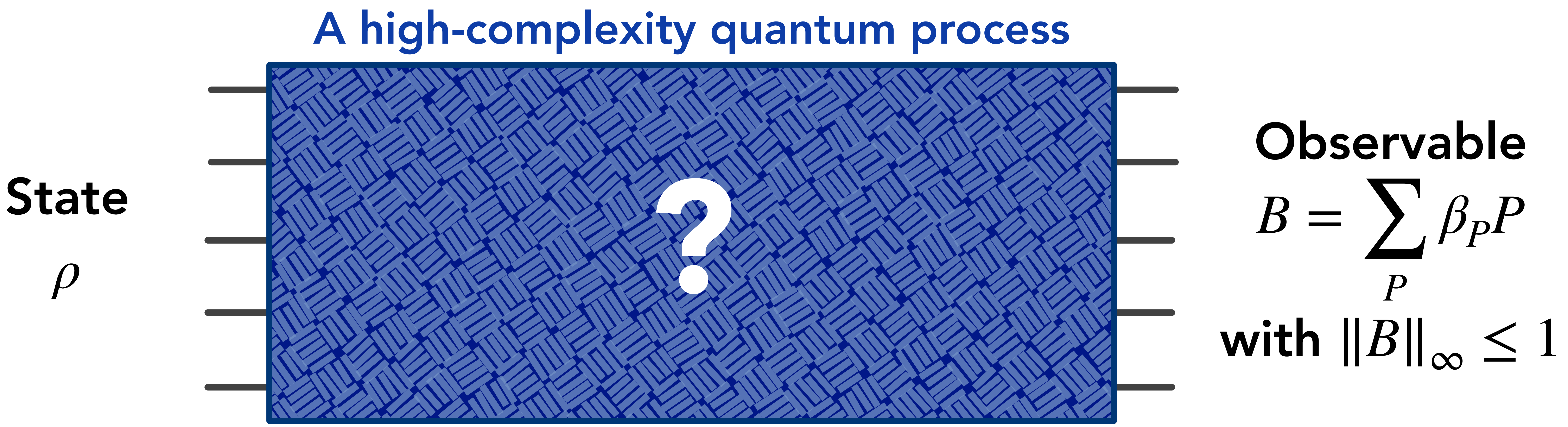
for $\ell = 1,\ldots,N.$

Properties [HKP20]:

$$\mathbb{E}[y_\ell] = \mathrm{Tr}(B\mathscr{E}(|\psi_\ell\rangle\langle\psi_\ell|))$$

$$\mathrm{Var}[y_\ell] \leq \|B\|^2_{\mathrm{shadow}}$$

Construct a dataset with classical shadow



For any sum of local observables B, $\|B\|_{\text{shadow}} \leq \mathcal{O}(\|\vec{\beta}\|_1) \leq \mathcal{O}(\|B\|_\infty)$ using the generalized quantum BH inequality.

$$|\psi_\ell\rangle = \bigotimes_{i=1} |\psi_{\ell,i}\rangle \mapsto y_\ell = \mathrm{Tr}\left(B \bigotimes_{i=1} \left(3|\phi_{\ell,i}\rangle\langle\phi_{\ell,i}| - I\right) \right)$$

for $\ell = 1, \ldots, N$.

$$\mathbb{E}[y_\ell] = \mathrm{Tr}(B\mathcal{E}(|\psi_\ell\rangle\langle\psi_\ell|))$$

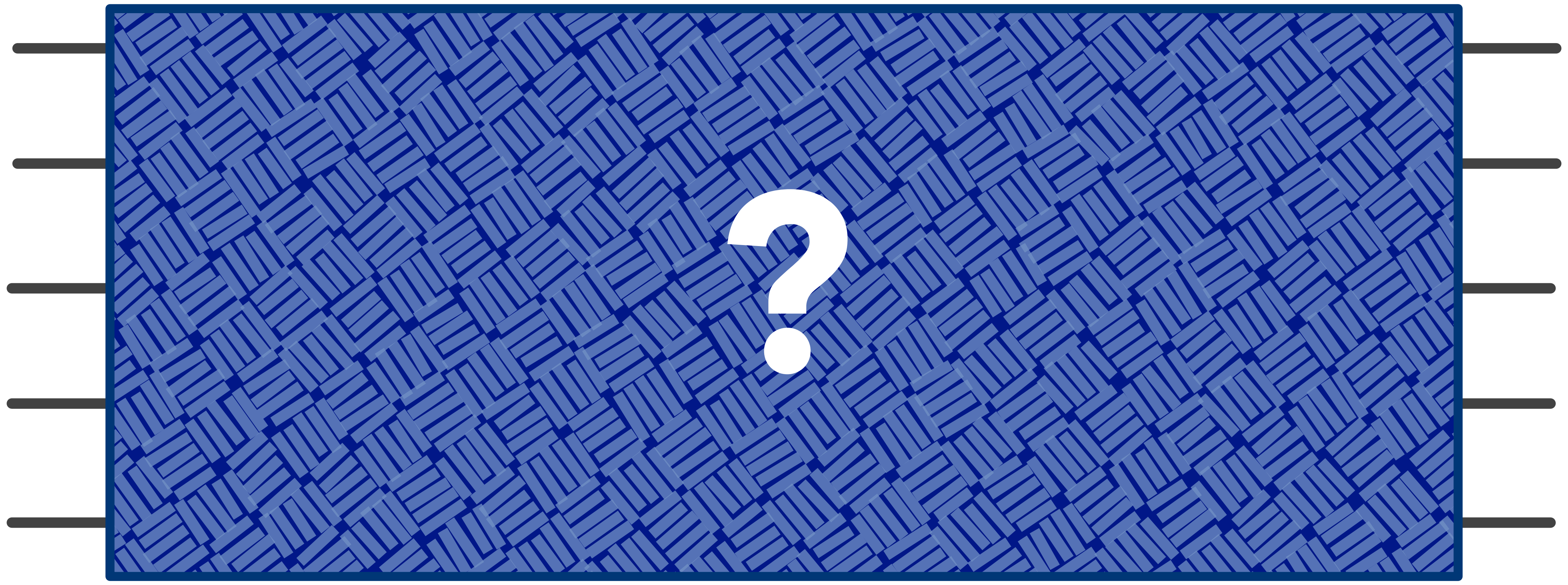
$$\mathrm{Var}[y_\ell] \leq \|B\|^2_{\text{shadow}}$$

Construct a dataset with classical shadow

A high-complexity quantum process

State

ρ



Observable

$$B = \sum_P \beta_P P$$

with $\|B\|_\infty \leq 1$

For any sum of local observables B, $\|B\|_{\text{shadow}} \leq \mathcal{O}(\|\vec{\beta}\|_1) \leq \mathcal{O}(\|B\|_\infty)$ using the generalized quantum BH inequality.

$$|\psi_\ell\rangle = \bigotimes_{i=1} |\psi_{\ell,i}\rangle \mapsto y_\ell = \mathrm{Tr}\left(B \bigotimes_{i=1} \left(3|\phi_{\ell,i}\rangle\langle\phi_{\ell,i}| - I\right) \right)$$

for $\ell = 1, \ldots, N$.

$$\mathbb{E}[y_\ell] = \mathrm{Tr}(B\mathcal{E}(|\psi_\ell\rangle\langle\psi_\ell|))$$

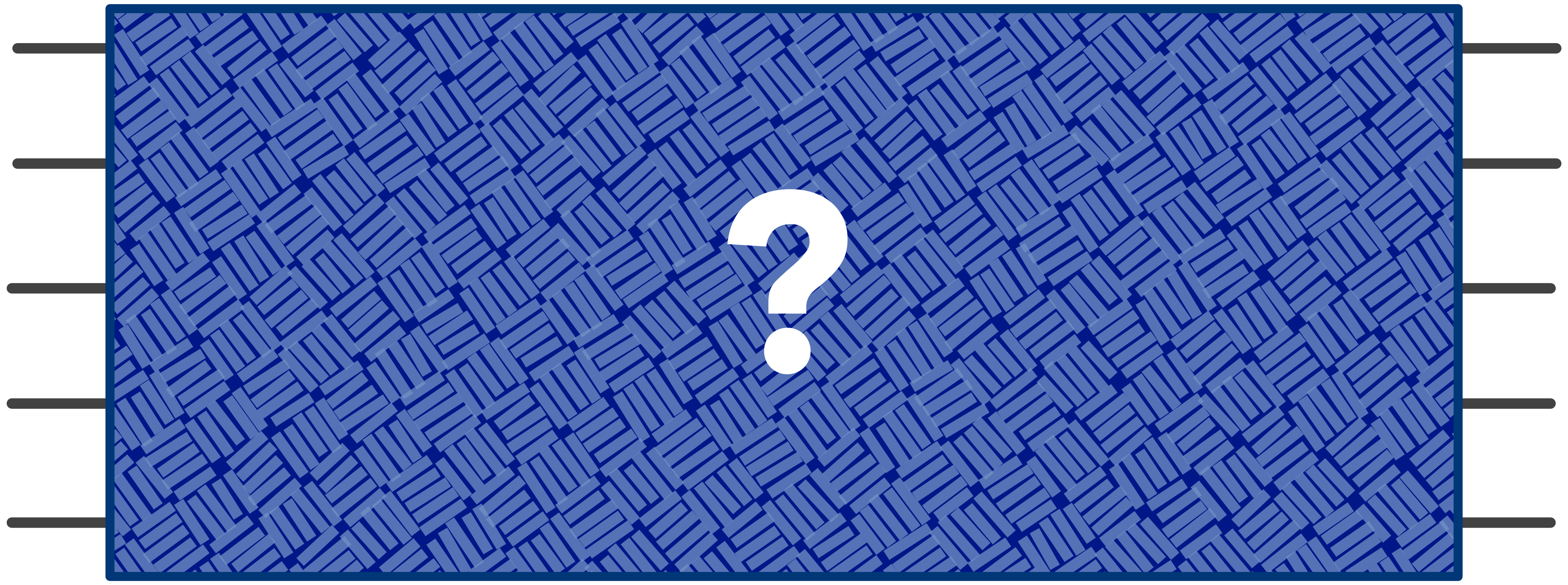
$$\mathrm{Var}[y_\ell] = \mathcal{O}(1)$$

Construct a dataset with classical shadow

A high-complexity quantum process

State

ρ



Observable

$$B = \sum_P \beta_P P$$

with $\|B\|_\infty \leq 1$

A New Classical Dataset

$$|\psi_\ell\rangle = \bigotimes_{i=1}^{n} |\psi_{\ell,i}\rangle \mapsto y_\ell = \mathrm{Tr}\left(B \bigotimes_{i=1}^{n} \left(3|\phi_{\ell,i}\rangle\langle\phi_{\ell,i}| - I\right) \right)$$

for $\ell = 1, \ldots, N.$

Properties [HKP20]:

$$\mathbb{E}[y_\ell] = \mathrm{Tr}(B\mathscr{E}(|\psi_\ell\rangle\langle\psi_\ell|))$$

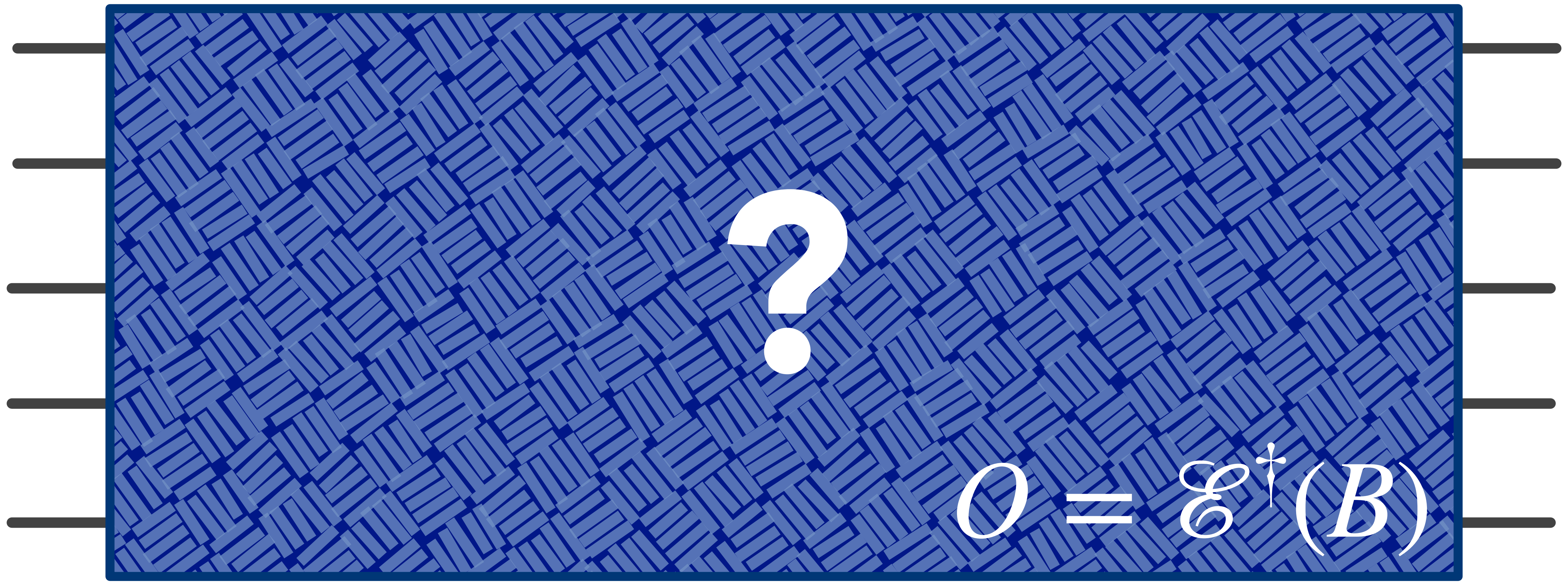
$$\mathrm{Var}[y_\ell] = \mathcal{O}(1)$$

Almost back to the previous problem

A high-complexity quantum process

State

ρ



Observable

$$B = \sum_P \beta_P P$$

with $\|B\|_\infty \leq 1$

Classical Dataset for O

$$|\psi_\ell\rangle = \bigotimes_{i=1}^{n} |\psi_{\ell,i}\rangle \mapsto y_\ell, \quad \mathbb{E}[y_\ell] = \langle \psi_\ell | O | \psi_\ell \rangle$$

for $\ell = 1, \ldots, N$.

Low-weight approximation

$$O = \sum_{P \in \{I,X,Y,Z\}^{\otimes n}} \alpha_P P$$

$$O^{(\text{low})} = \sum_{|P| \leq k} \alpha_P P$$

Lemma (Low-weight approximation): $\mathbb{E}_{\rho \sim \mathcal{D}} \left| \mathrm{Tr}(O\rho) - \mathrm{Tr}(O^{(\text{low})}\rho) \right|^2 < \frac{1}{1.5^k}.$

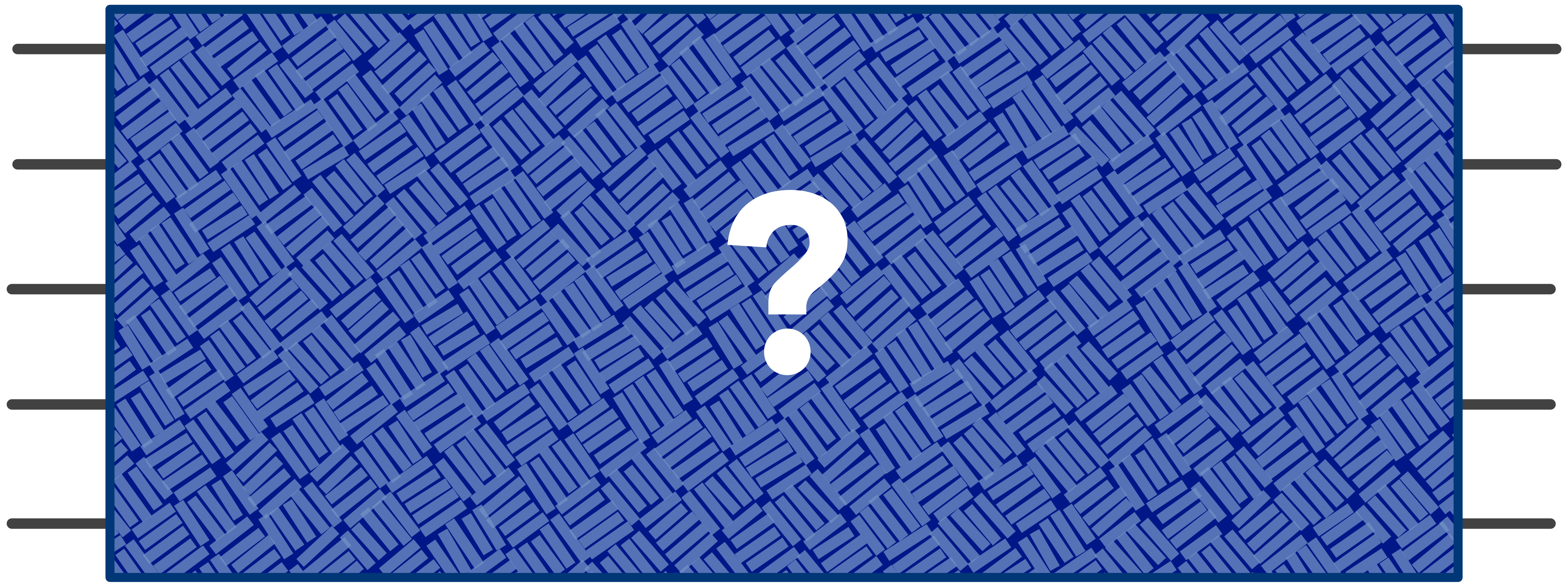
The lemma holds for any distribution $\mathcal{D}$ over any quantum state ρ

as long as $\mathcal{D}$ is flat under single-qubit rotations.

Example: ρ is the ground/thermal state of a generic geometrically-local Hamiltonian.

The ML algorithm

A high-complexity quantum process



State

ρ

Observable

$B = \sum_P \beta_P P$

with $\|B\|_\infty \leq 1$

Classical Dataset

$$|\psi_\ell\rangle = \bigotimes_{i=1}^{n} |\psi_{\ell,i}\rangle \mapsto |\phi_\ell\rangle = \bigotimes_{i=1}^{n} |\phi_{\ell,i}\rangle$$

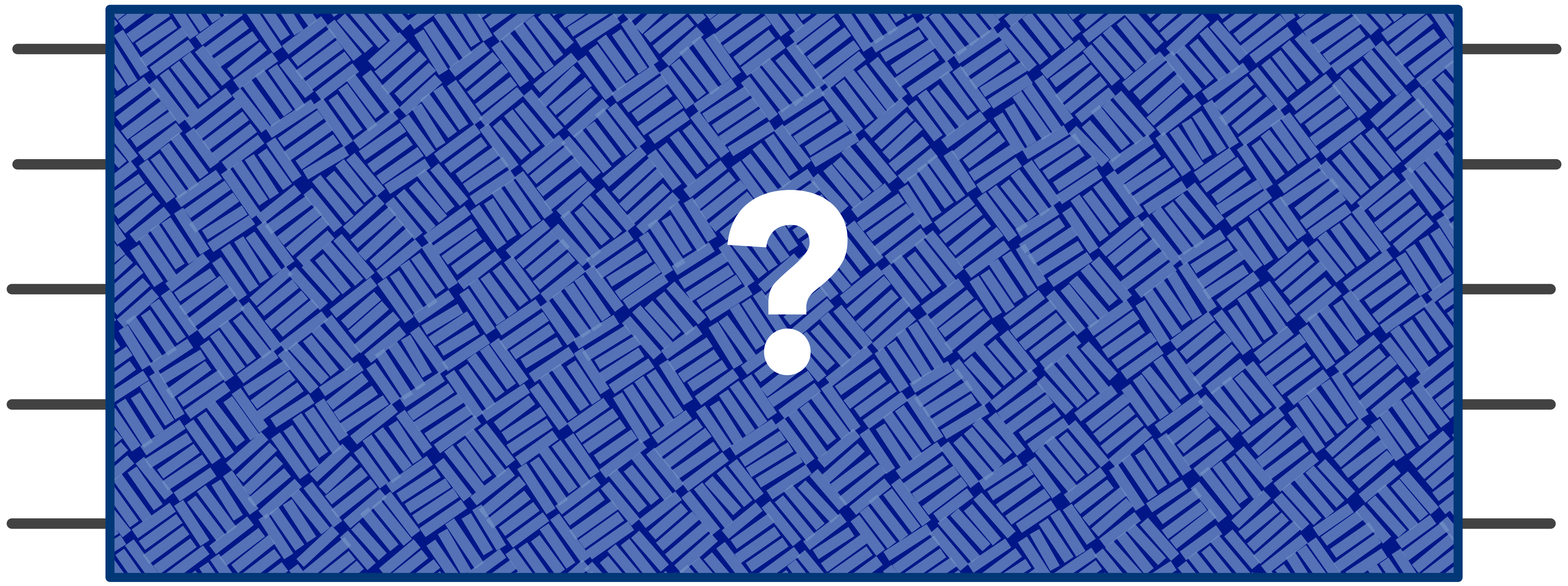
for $\ell = 1, \ldots, N$.

The ML algorithm

A high-complexity quantum process

State

ρ



Observable

$B = \sum_P \beta_P P$

with $\|B\|_\infty \leq 1$

Classical Dataset for $O = \mathscr{E}^\dagger(B)$

$$|\psi_\ell\rangle = \bigotimes_{i=1}^{n} |\psi_{\ell,i}\rangle \mapsto y_\ell, \quad \mathbb{E}[y_\ell] = \langle \psi_\ell | O | \psi_\ell \rangle$$

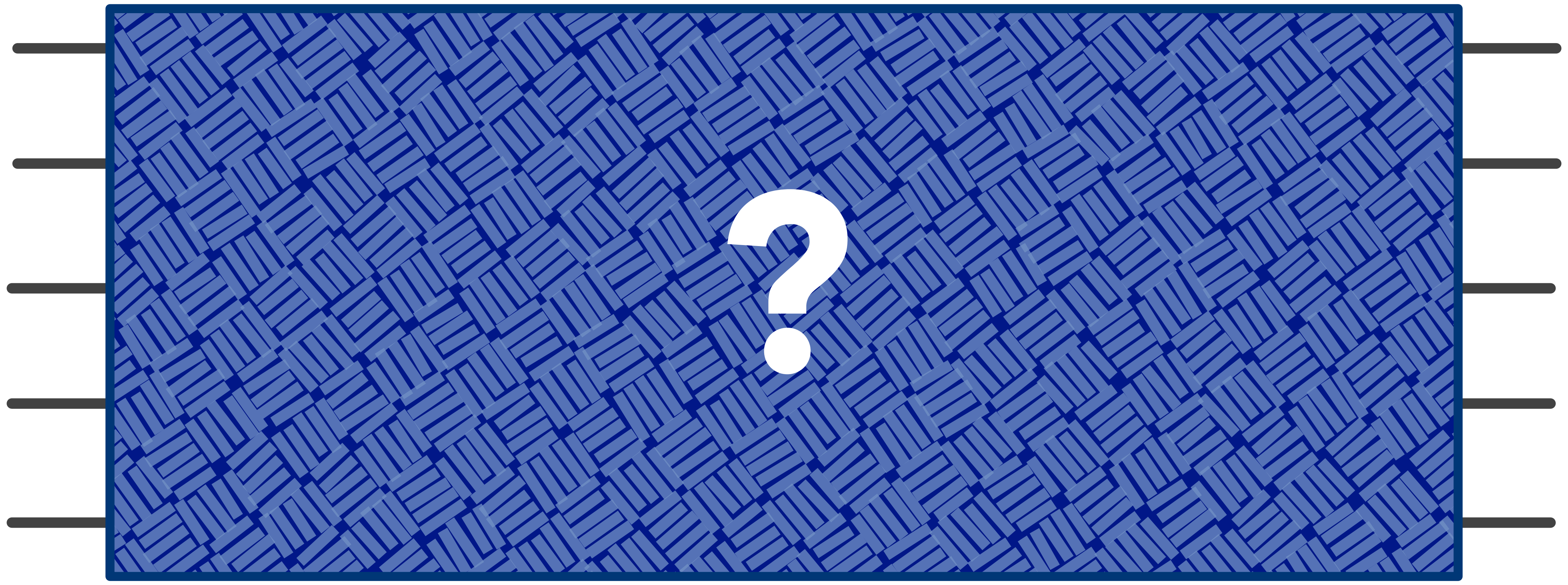
for $\ell = 1, \ldots, N$.

The ML algorithm

A high-complexity quantum process

State

ρ



Observable

$$B = \sum_P \beta_P P$$

with $\|B\|_\infty \leq 1$

Classical Dataset for $O = \mathscr{E}^\dagger(B)$

$$|\psi_\ell\rangle = \bigotimes_{i=1}^{n} |\psi_{\ell,i}\rangle \mapsto y_\ell, \quad \mathbb{E}[y_\ell] = \langle\psi_\ell|O|\psi_\ell\rangle$$

for $\ell = 1, \ldots, N$.

For all $|P| \leq k$,

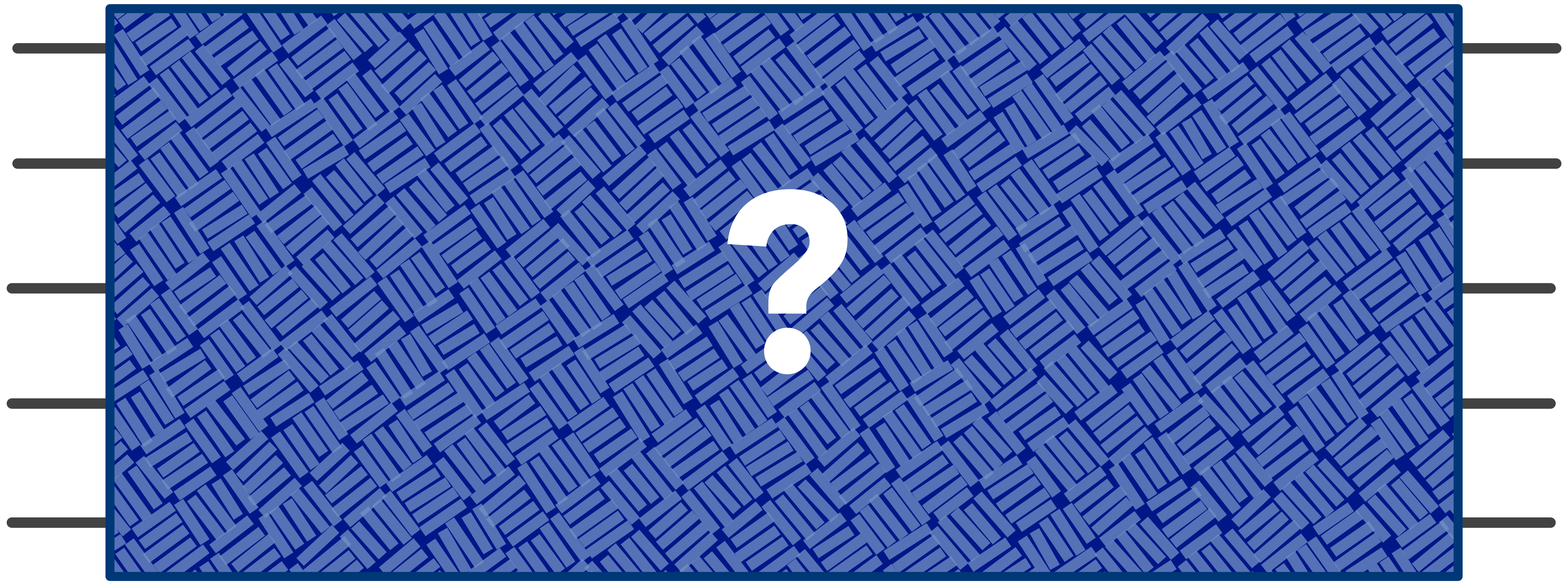
set $\hat{\alpha}_P \leftarrow \frac{3^{|P|}}{N} \sum_{\ell=1}^{N} y_\ell \langle\psi_\ell|P|\psi_\ell\rangle$.

If $\hat{\alpha}_P$ is small, set $\hat{\alpha}_P \leftarrow 0$.

The learned observable is $\hat{O}^{(\mathrm{low})} = \sum_{|P| \leq k} \hat{\alpha}_P P$.

The ML algorithm

A high-complexity quantum process



State

ρ

Observable

$$B = \sum_P \beta_P P$$

with $\|B\|_\infty \leq 1$

Classical Dataset for $O = \mathscr{E}^\dagger(B)$

$$|\psi_\ell\rangle = \bigotimes_{i=1}^{n} |\psi_{\ell,i}\rangle \mapsto y_\ell, \quad \mathbb{E}[y_\ell] = \langle \psi_\ell | O | \psi_\ell \rangle$$

for $\ell = 1, \ldots, N$.

Predict $\mathrm{Tr}\left(\hat{O}^{(\mathrm{low})}\rho\right) \approx \mathrm{Tr}\left(B\mathscr{E}(\rho)\right)$

Surprising aspects of the ML algorithm

- We can learn to predict n-qubit exponential-size quantum circuits up to a const. relative error from only $\mathcal{O}(\log n)$ samples.

Surprising aspects of the ML algorithm

- We can learn to predict n-qubit exponential-size quantum circuits up to a const. relative error from only $\mathcal{O}(\log n)$ samples.
- The algorithm is computationally efficient (polynomial time for a const. relative error; quasi-polynomial time for a small error).

Surprising aspects of the ML algorithm

- We can learn to predict n-qubit exponential-size quantum circuits up to a const. relative error from only $\mathcal{O}(\log n)$ samples.
- The algorithm is computationally efficient (polynomial time for a const. relative error; quasi-polynomial time for a small error).
- After learning from product state inputs, the algorithm can predict entangled states.

Surprising aspects of the ML algorithm

- We can learn to predict n-qubit exponential-size quantum circuits up to a const. relative error from only $\mathcal{O}(\log n)$ samples.
- The algorithm is computationally efficient (polynomial time for a const. relative error; quasi-polynomial time for a small error).
- After learning from product state inputs, the algorithm can predict entangled states.
- The entire algorithm can be run on a classical computer.

Conclusion

- We give a computationally-efficient ML algorithm that can learn to predict the output of a quantum process with arbitrary complexity.
- Our results highlight the potential that ML models can predict outcomes of a complex quantum dynamics much faster than the process itself.