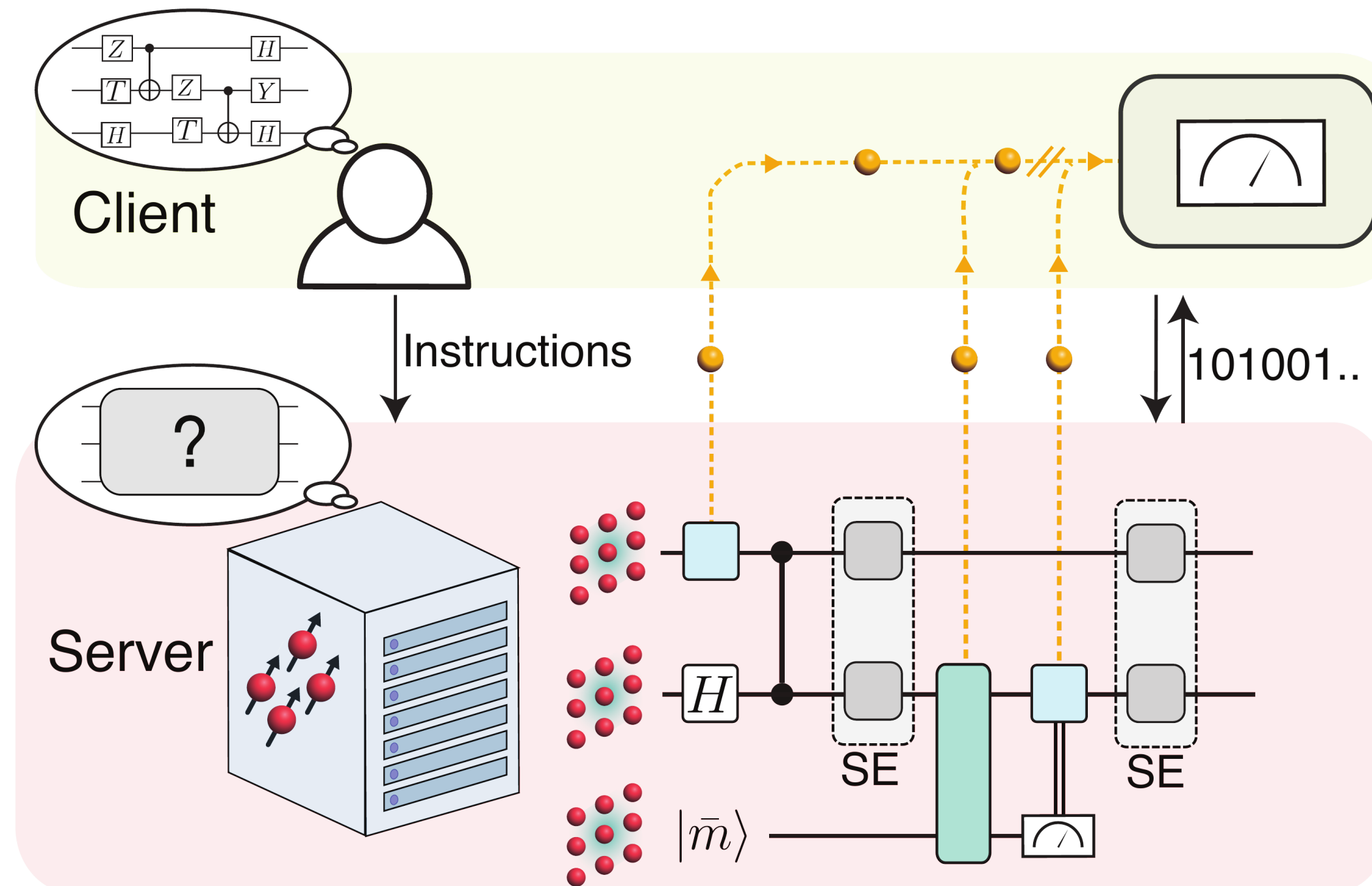


Designing Fault-Tolerant Delegated Blind Computation

Francisco Machado @ QuTech



Summit Quantum Limits @ Dokkum
Sep 7 2026

Gefen Baranes Iria Wang



Yan-Cheng
Wei



Pieter-Jan
Stas



Aziza
Suleymanzade



Susanne
Yelin



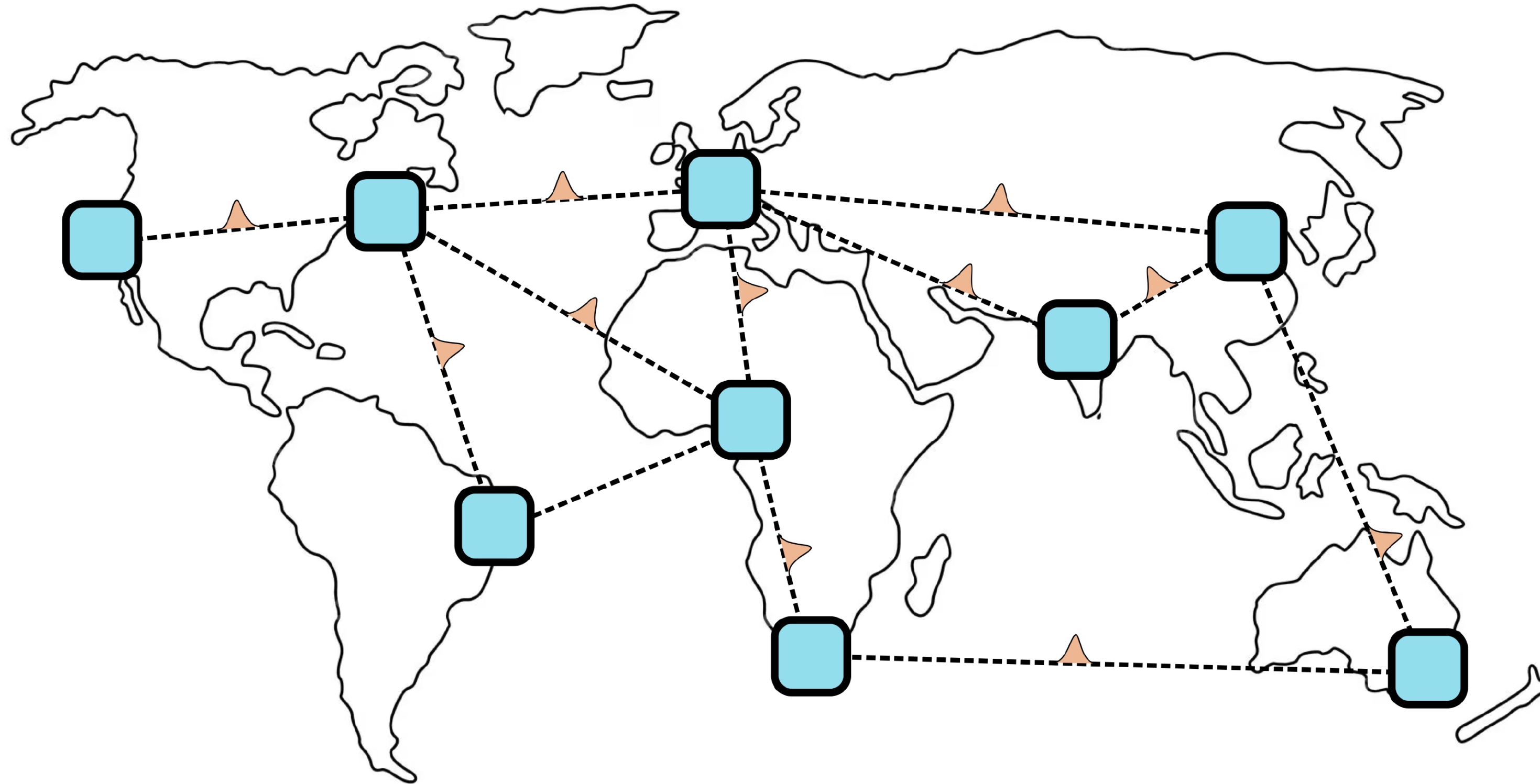
Johannes
Borregaard



Mikhail
Lukin

arXiv:2505.21621

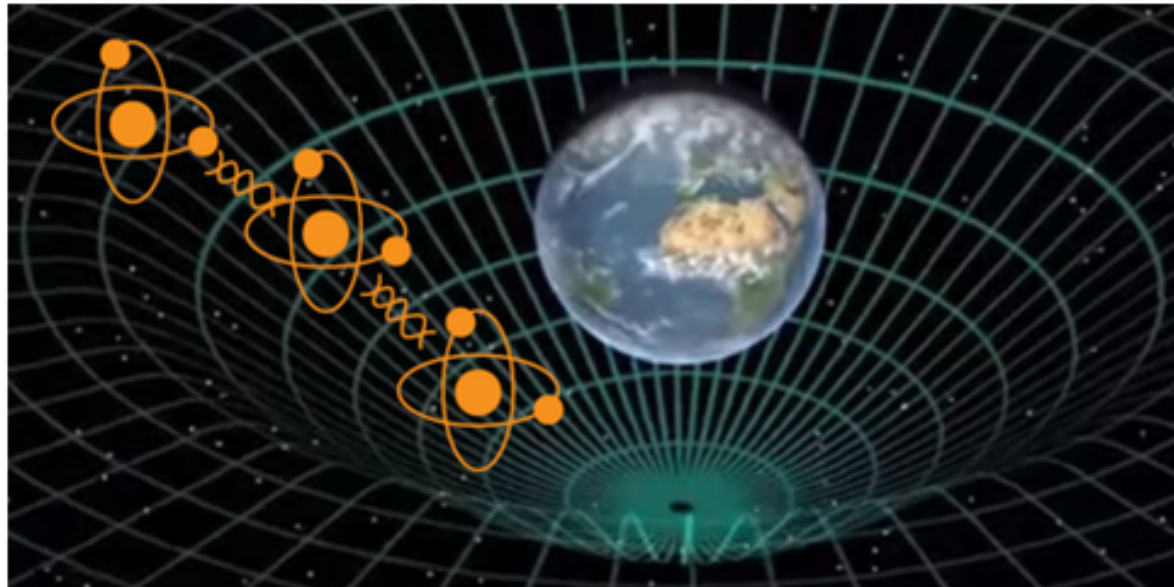
Quantum Entanglement at a Distance: a final frontier



Quantum Entanglement at a Distance: a final frontier

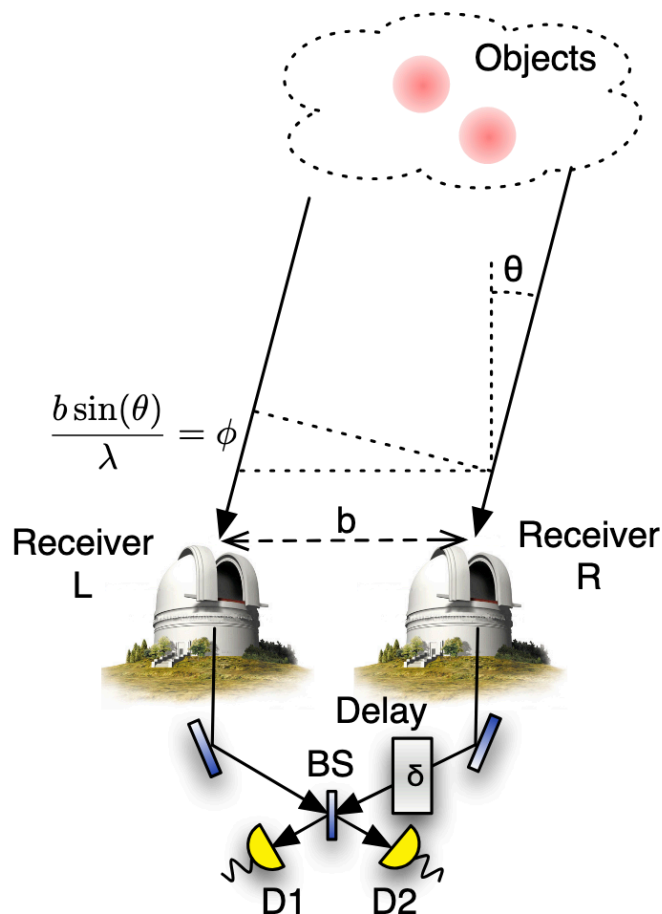
Science

Fundamental Science

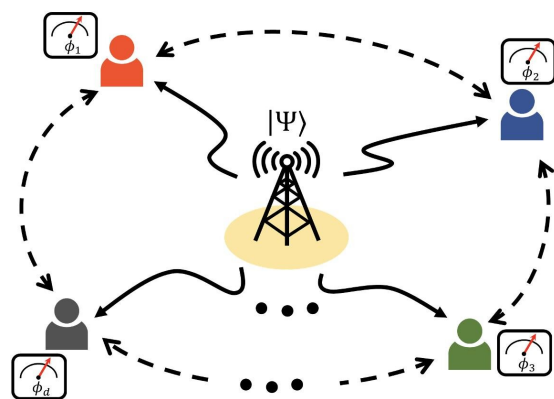


Covey PRX Quantum 2025

Distributed Sensing

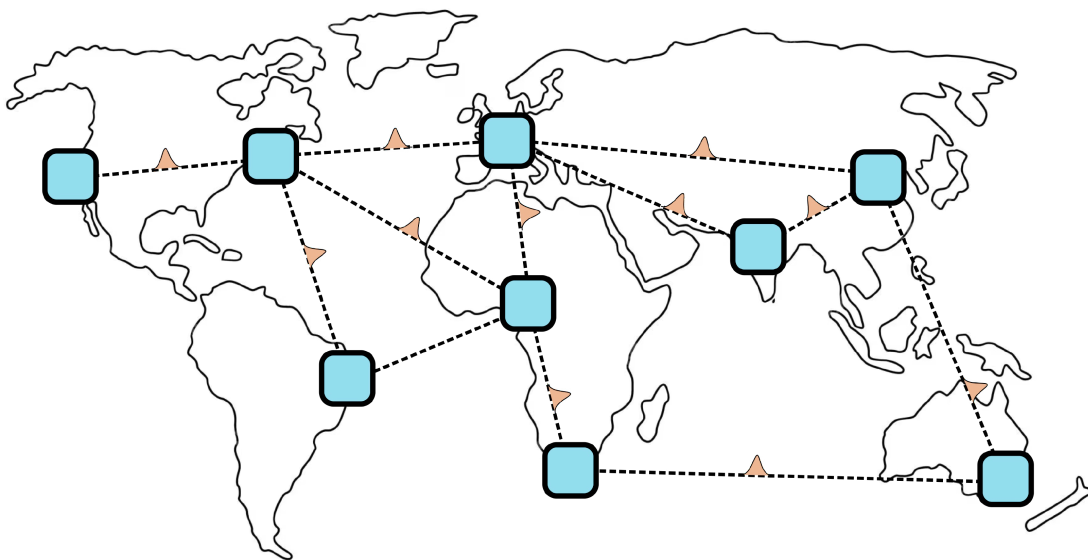


Gottesman et al PRL 2012



Kim et al Nature Comm 2023

Quantum Internet



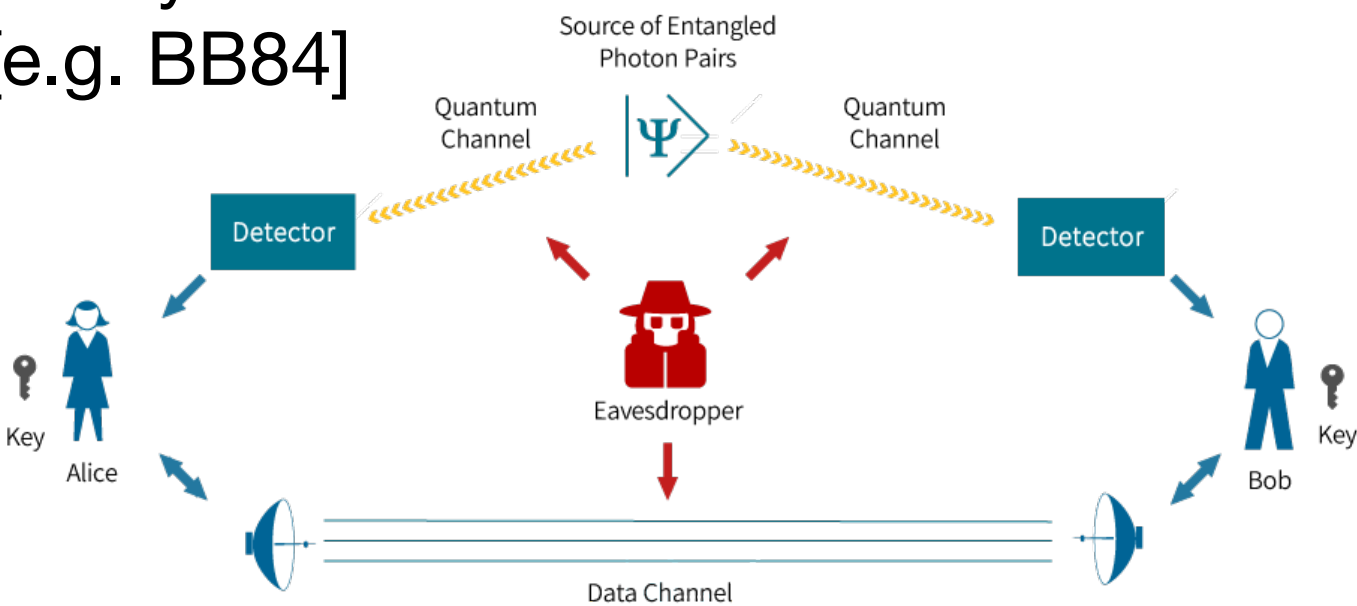
Kimble Nature 2008

Azuma et al Rev Mod Phys 2023

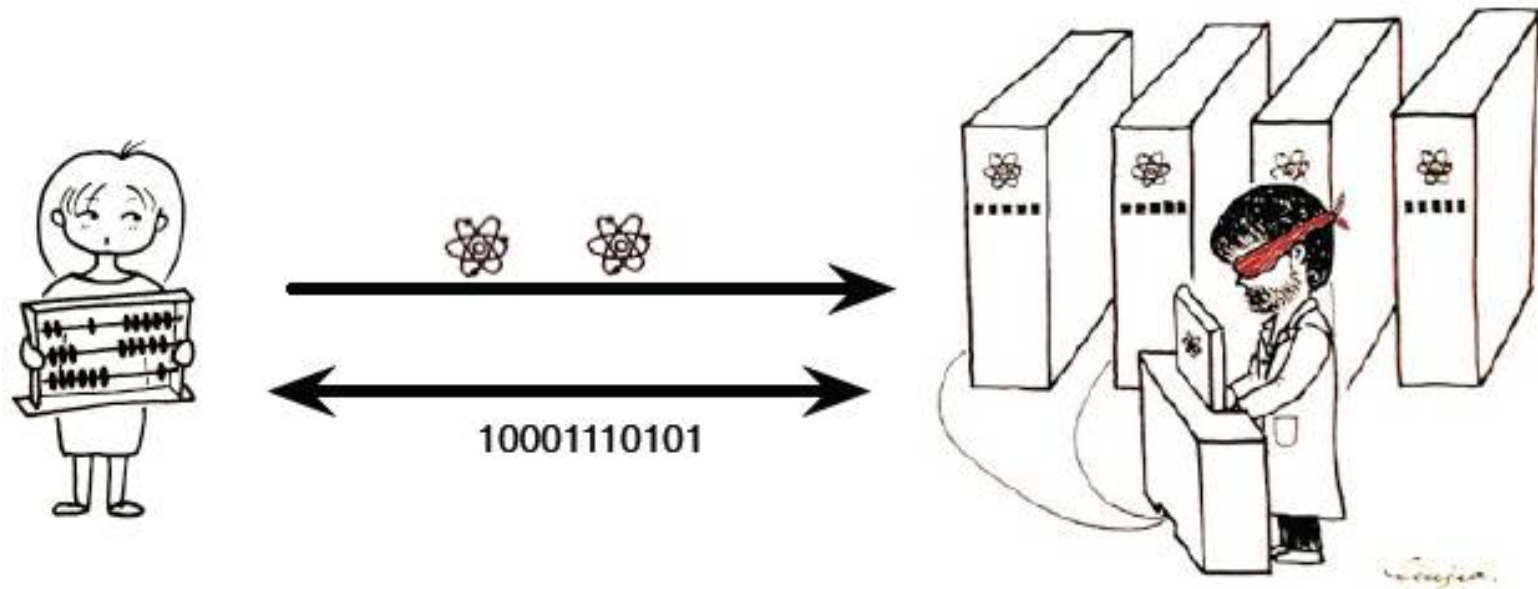
Technology

Secure Communication and Computation

Quantum Key Distribution [e.g. BB84]

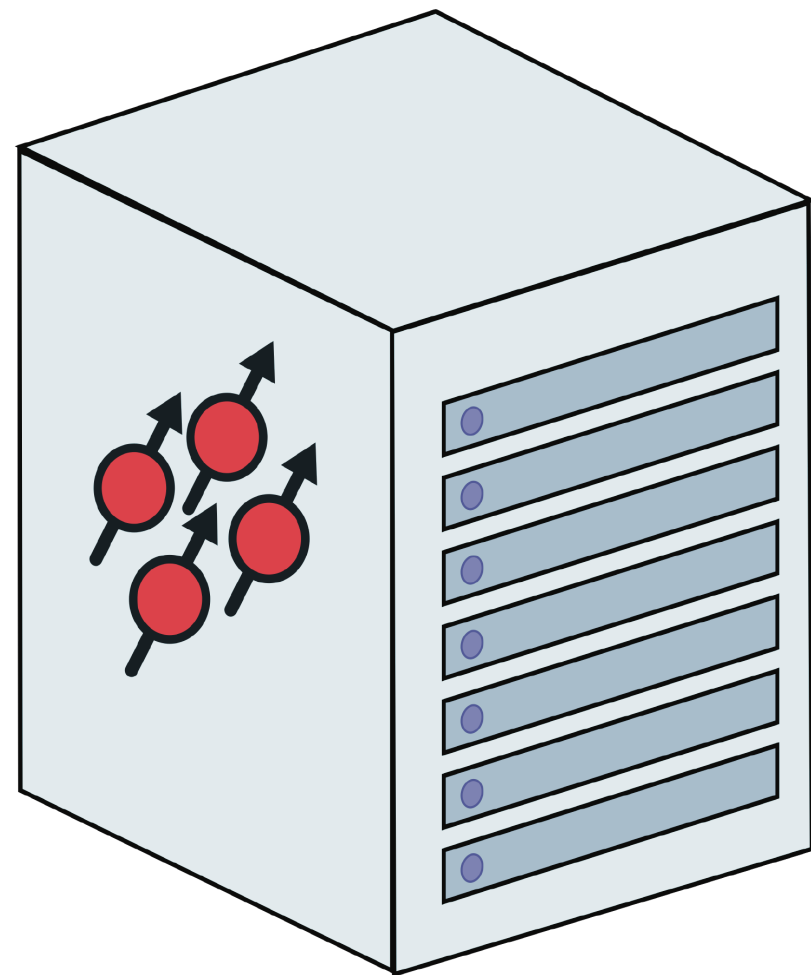


Delegated Secure Quantum Computation [i.e. Blind Quantum Computing]



Blind Quantum Computing (BQC)

Server



Capable of Fault-Tolerant QC

Client



Limited quantum resources

Wants to run a quantum algorithm
But does not trust the server
(i.e. wants to keep algorithm and data hidden)



Blind QC Protocol

Involving classical/quantum communication

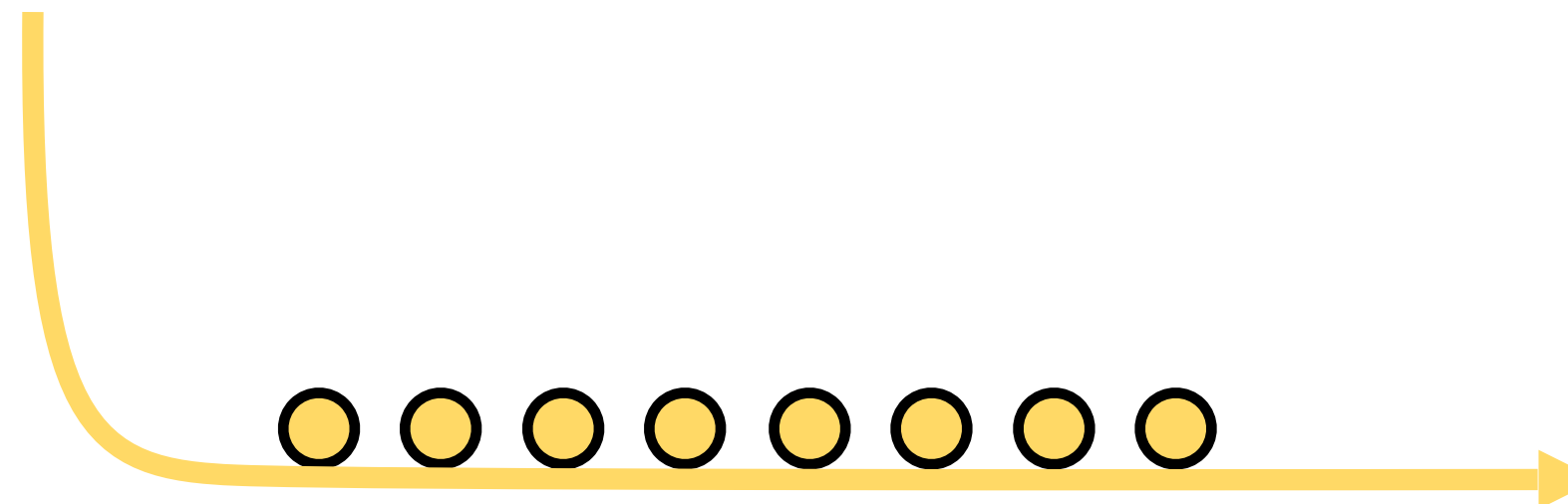
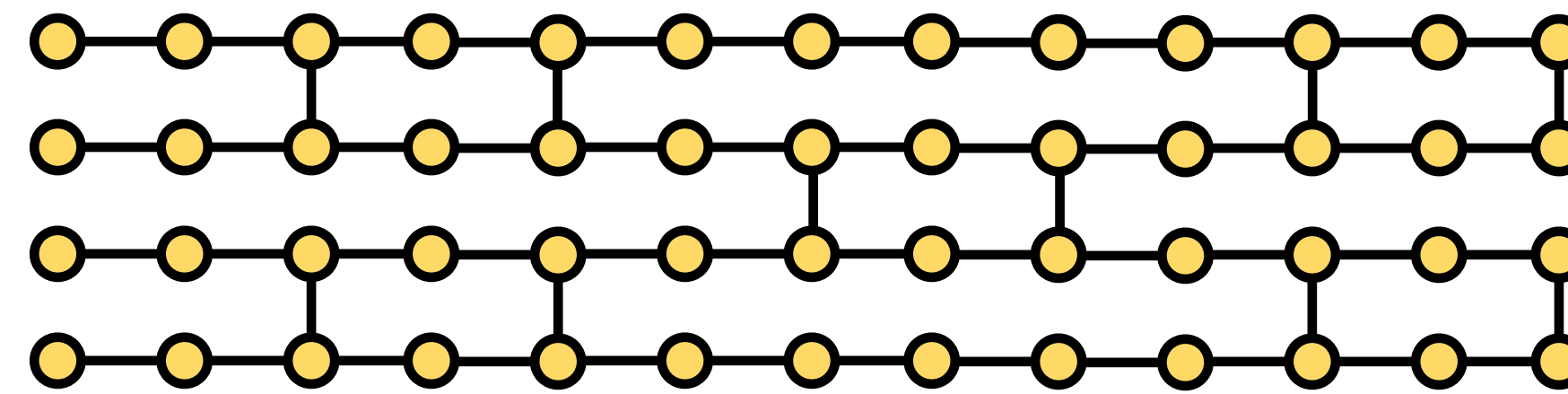
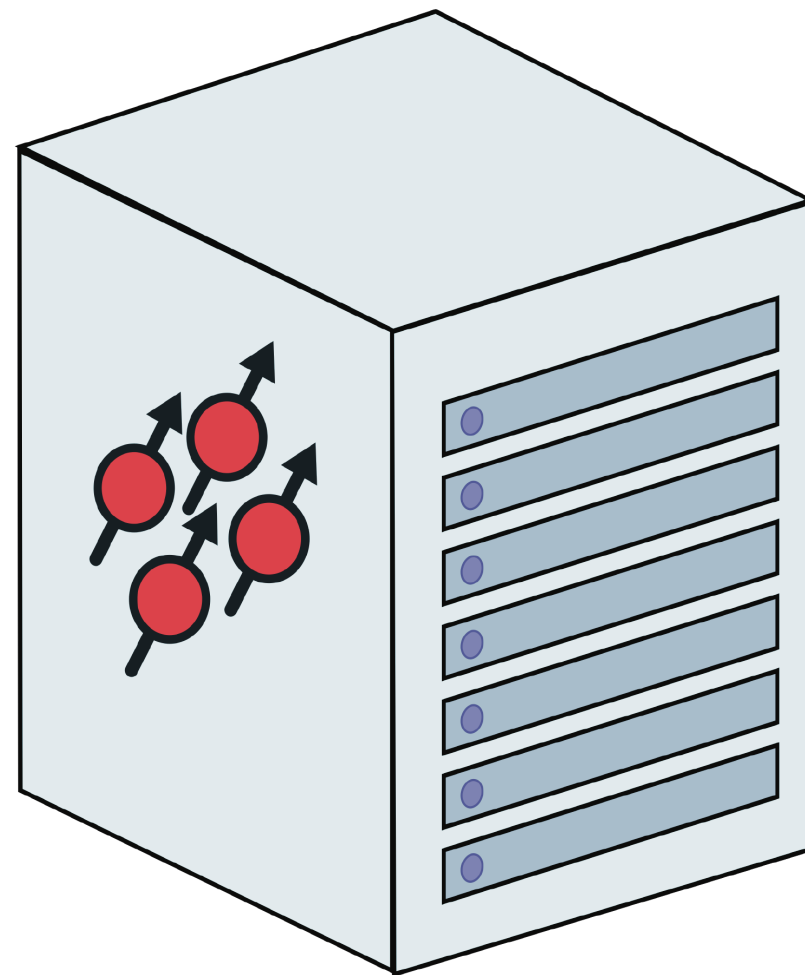
“Simplest approach”: Measurement-Based QC

Server

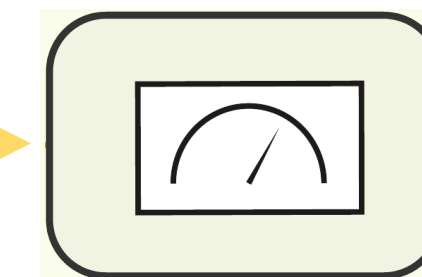
Uses quantum resources to
prepare a resource state

● photon

Client



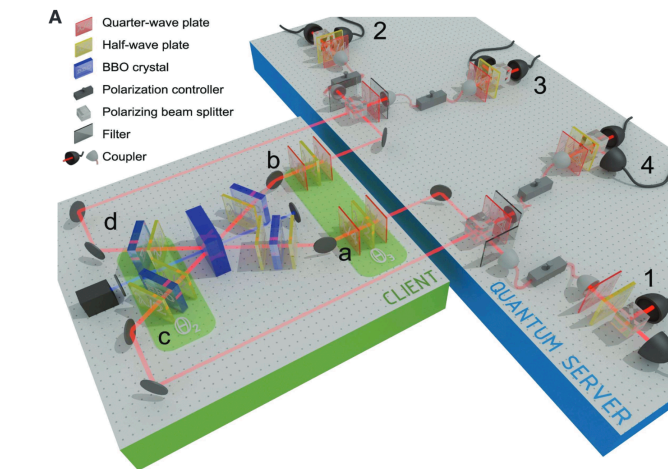
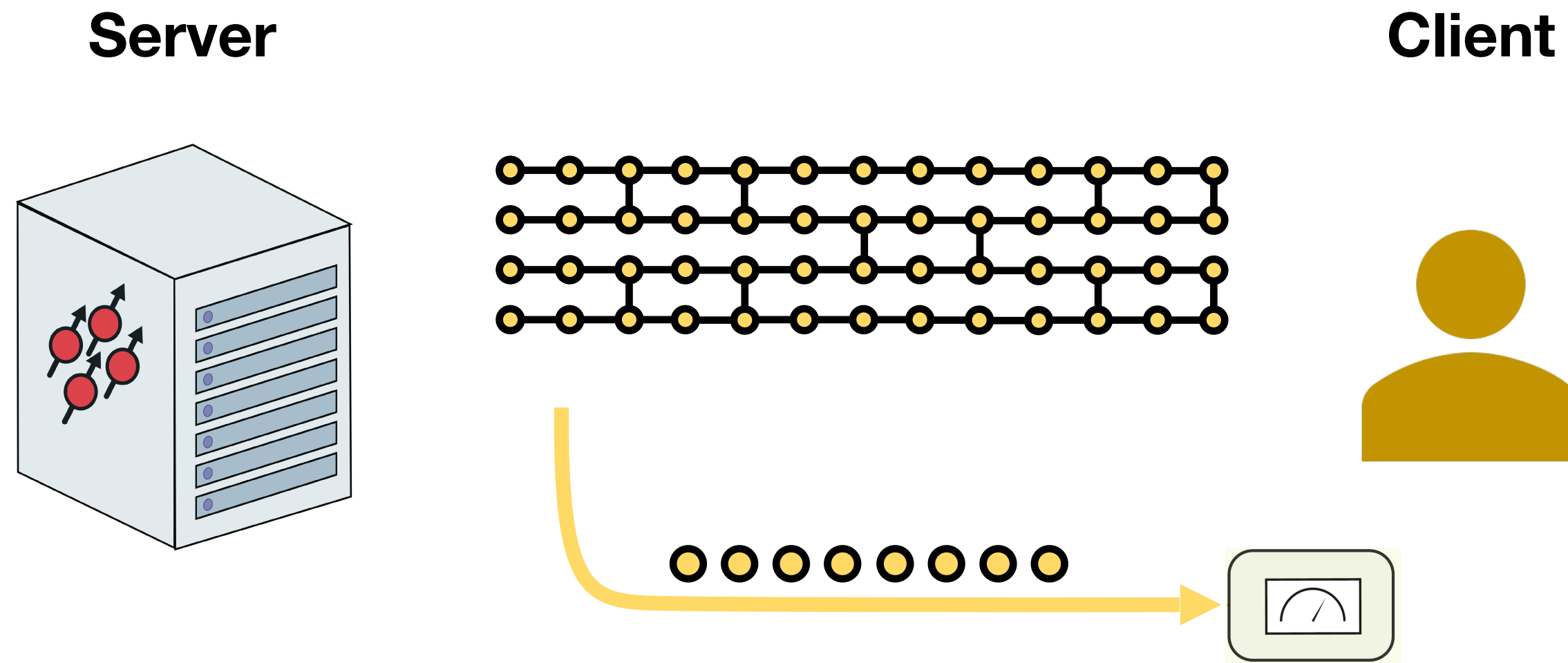
Which are sent to the client
and measured



Measurement pattern enables the
client to perform any computation
(given large enough resource state)

1-way Protocol: The server cannot
learn any details of the computation

Simple yet hard: Challenges for large-scale BQC



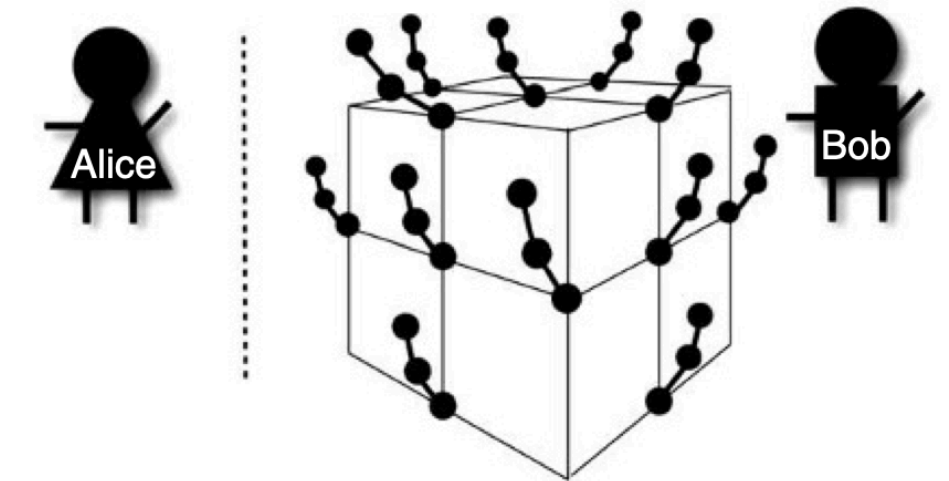
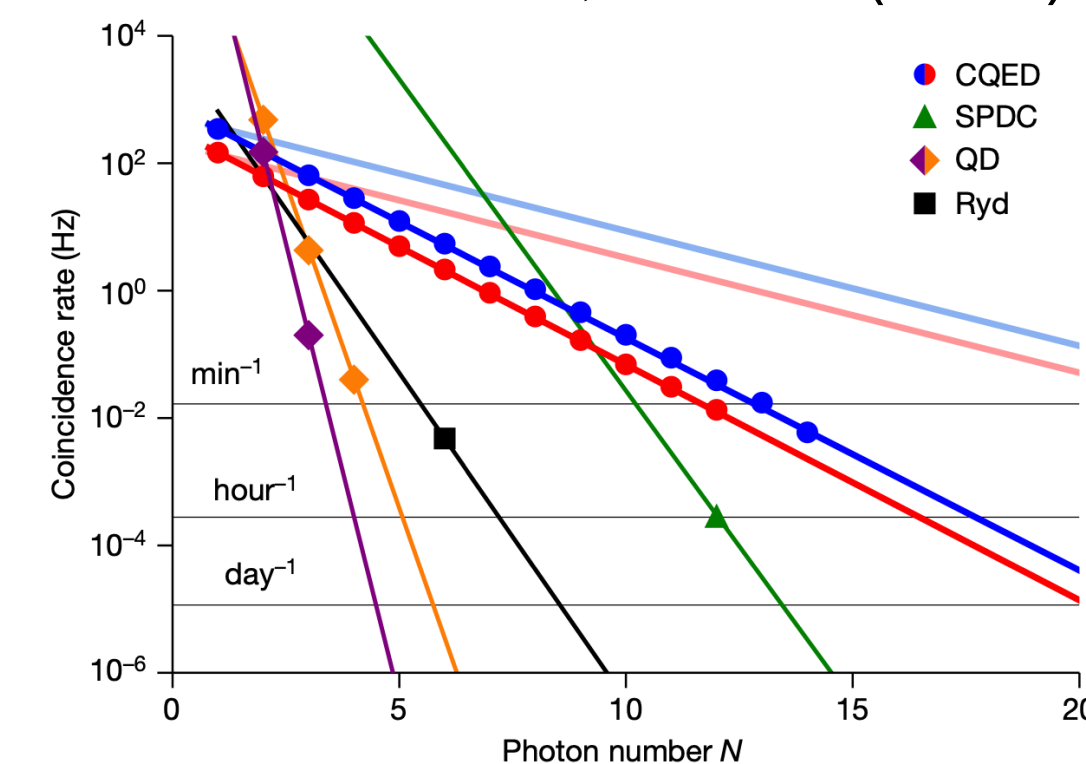
Photonic experimental BQC

S. Barz et al, Science (2012)

S. Barz et al, Nature Physics (2013)

Polacchi et al, arXiv:2407.09310 (2024)

P. Thomas et al, Nature (2022)



Topological BQC
Morimae and Fujii,
Nature Comm (2012)

Key Challenges:

- 1) Transmission $\Rightarrow$ photon loss
- 2) Errors / non-deterministic entanglement

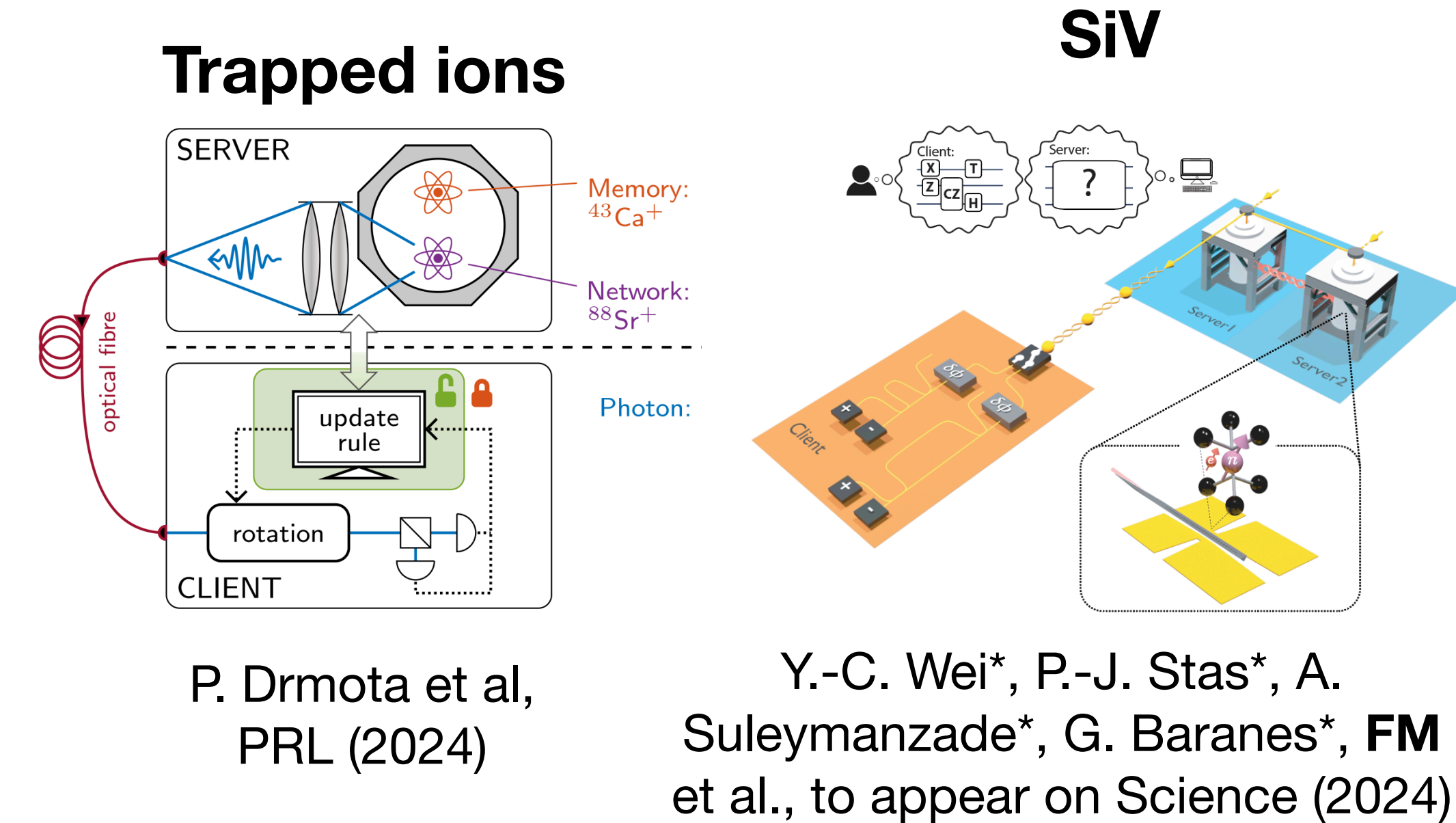
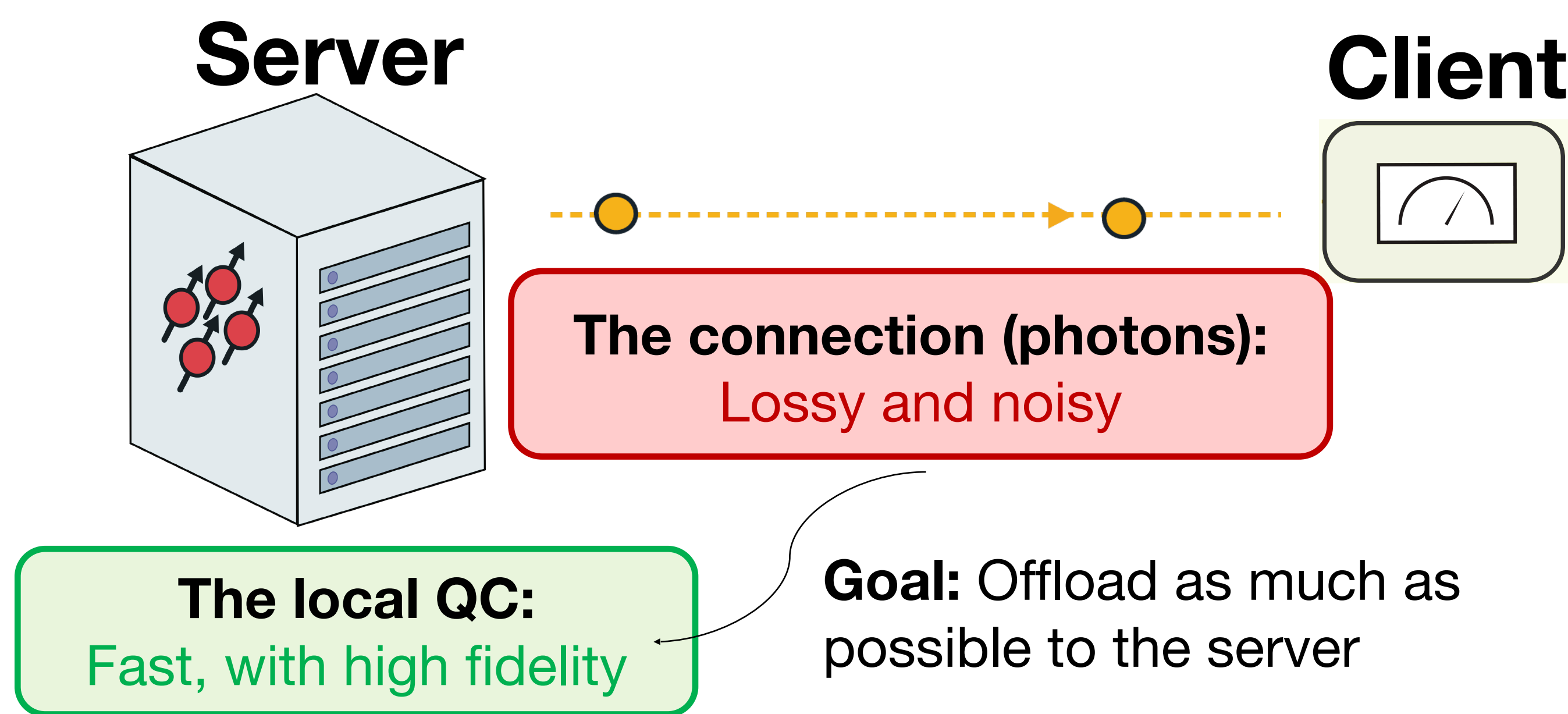
Answer: “Just” choose a better resource state

“Just” compile blind error correction

Meaning:

- More photons
- More complicated state topologies
- More overhead
- Very bad loss/error thresholds
(very stringent expt requirements)

New opportunities when thinking holistically



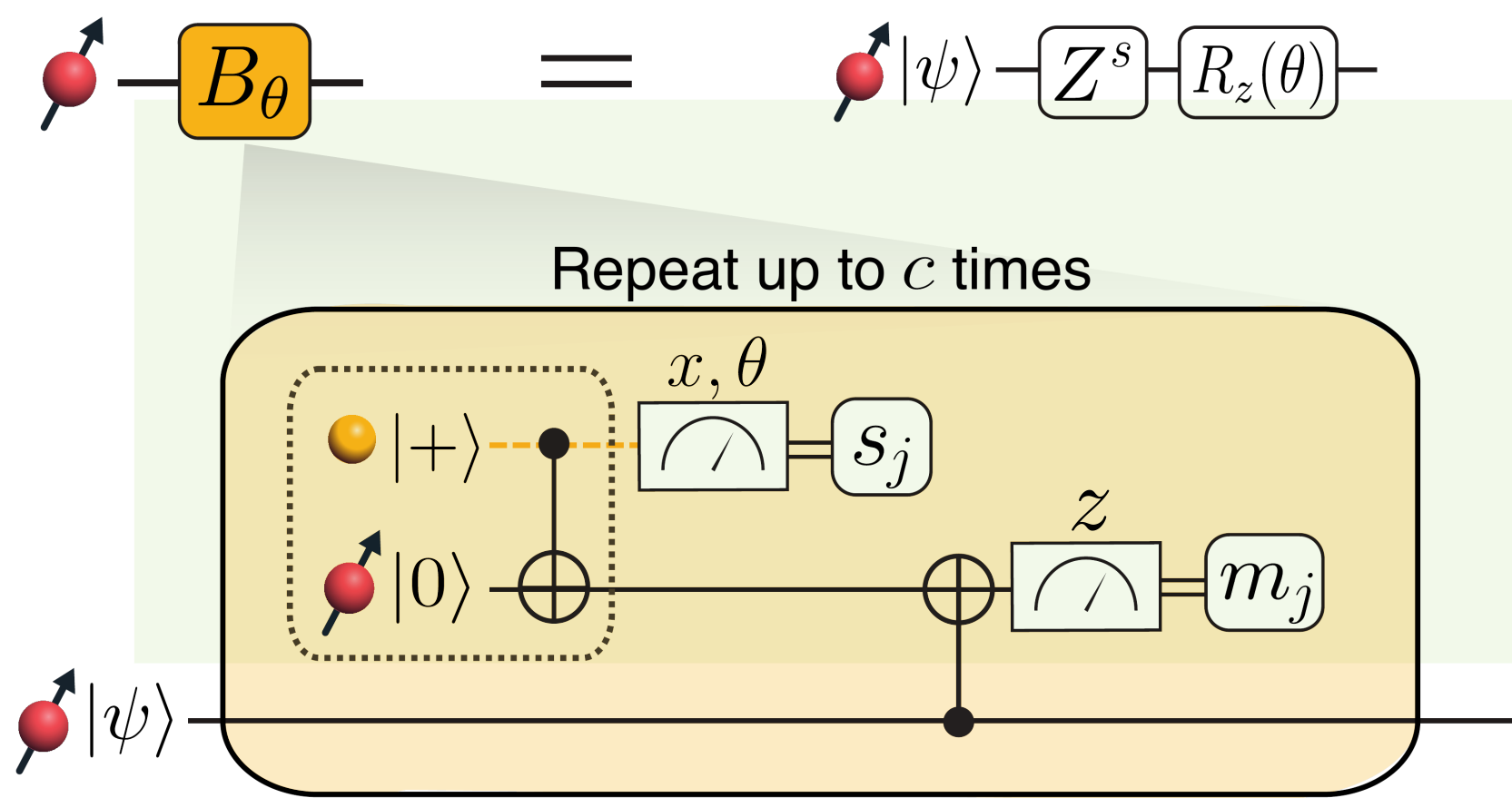
Server's matter qubits = robust quantum processor

- Deterministic generator of entanglement
- Memory qubits enables retry until success protocols

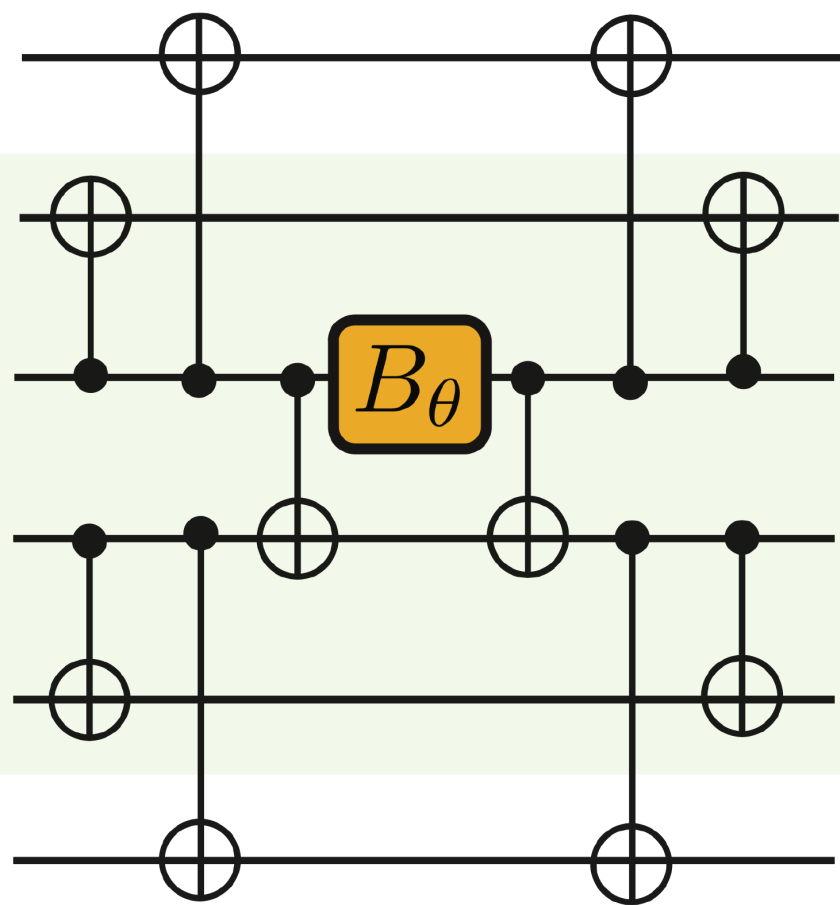
How much more can we gain if we think of the entire stack?

Designing Fault-Tolerant Blind Quantum Computing

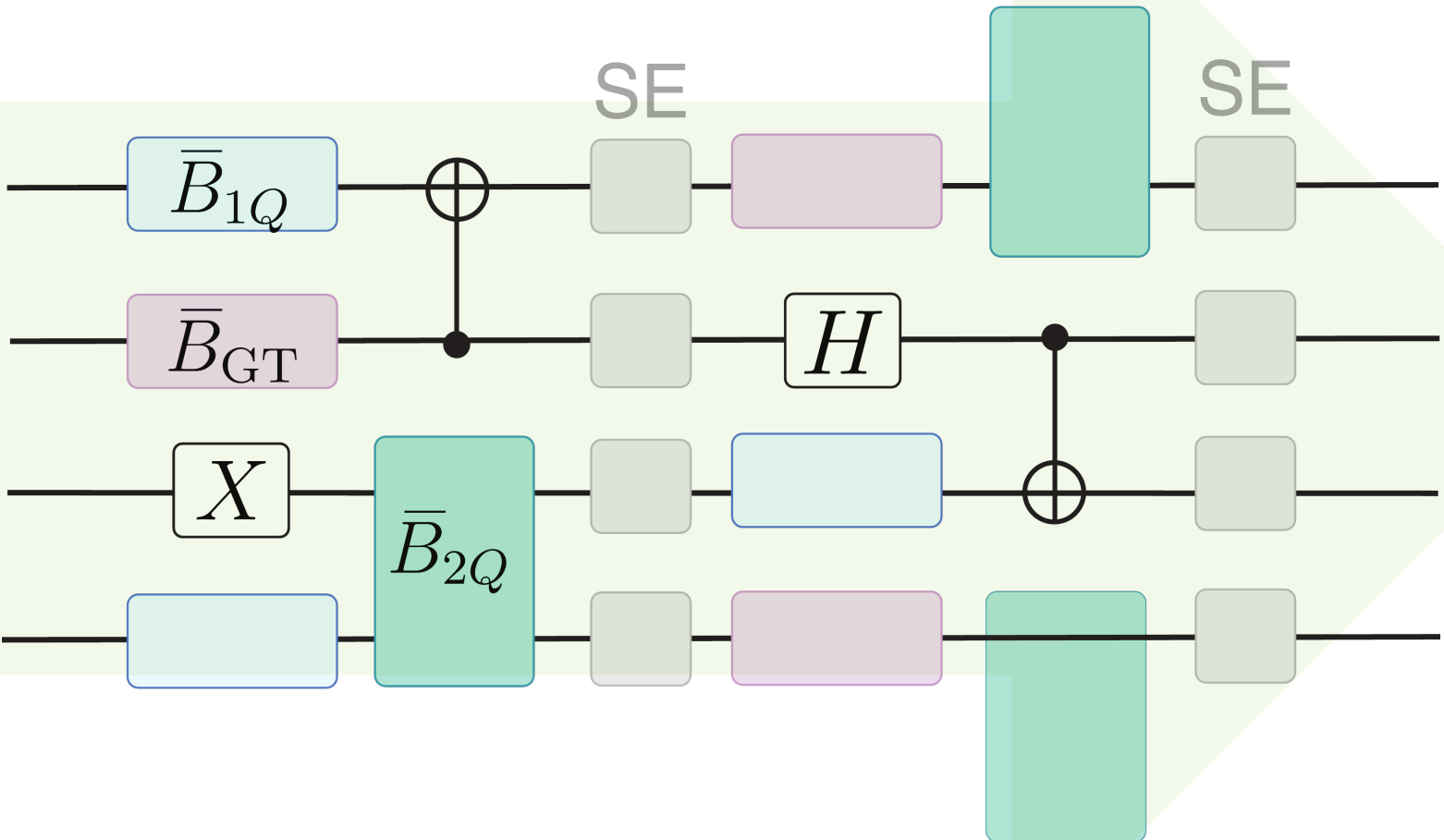
Hybrid Light-Matter Blind QC



Exploring new circuits design possibilities



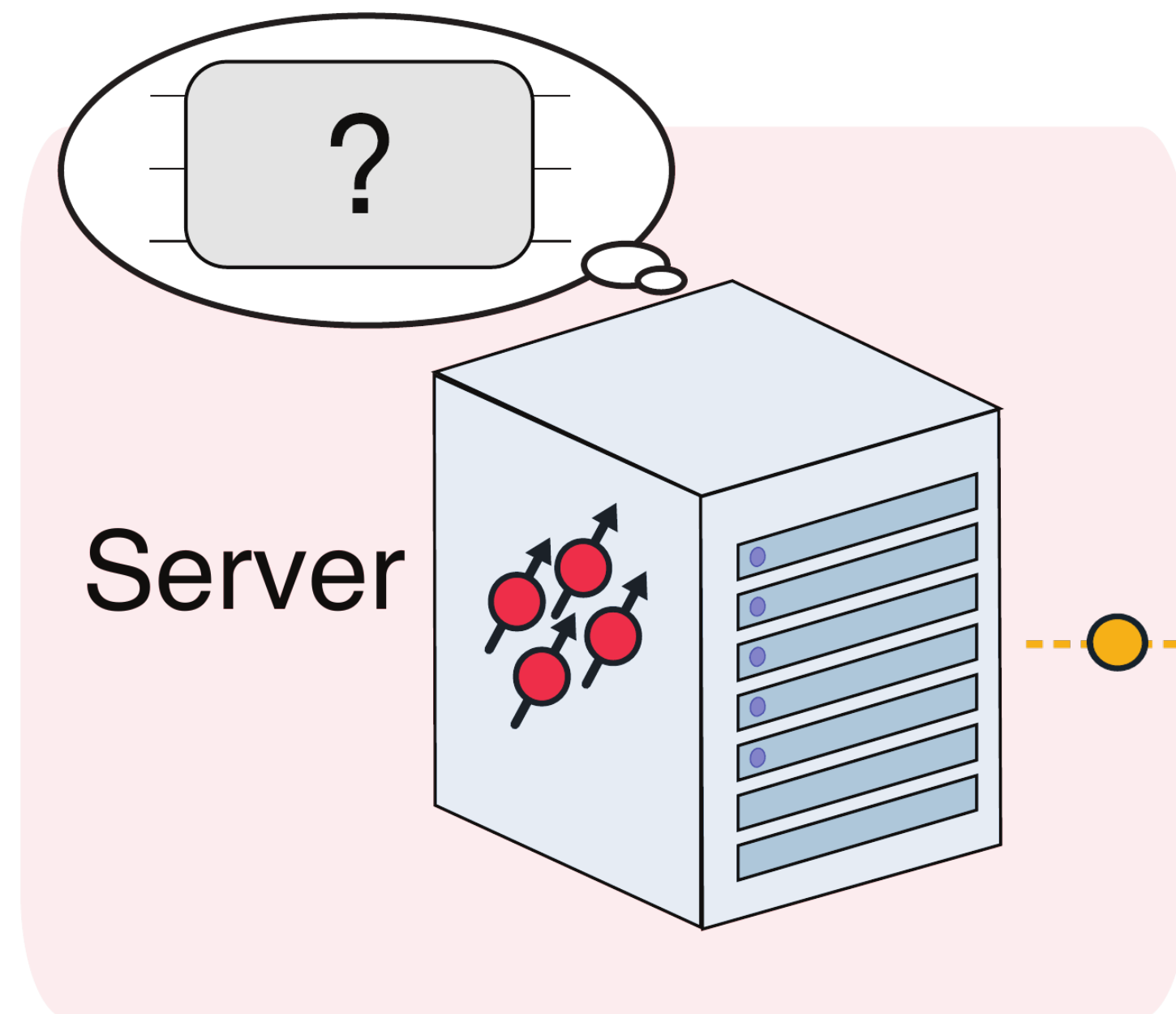
Fault-tolerant scalable implementation



Rules of the game

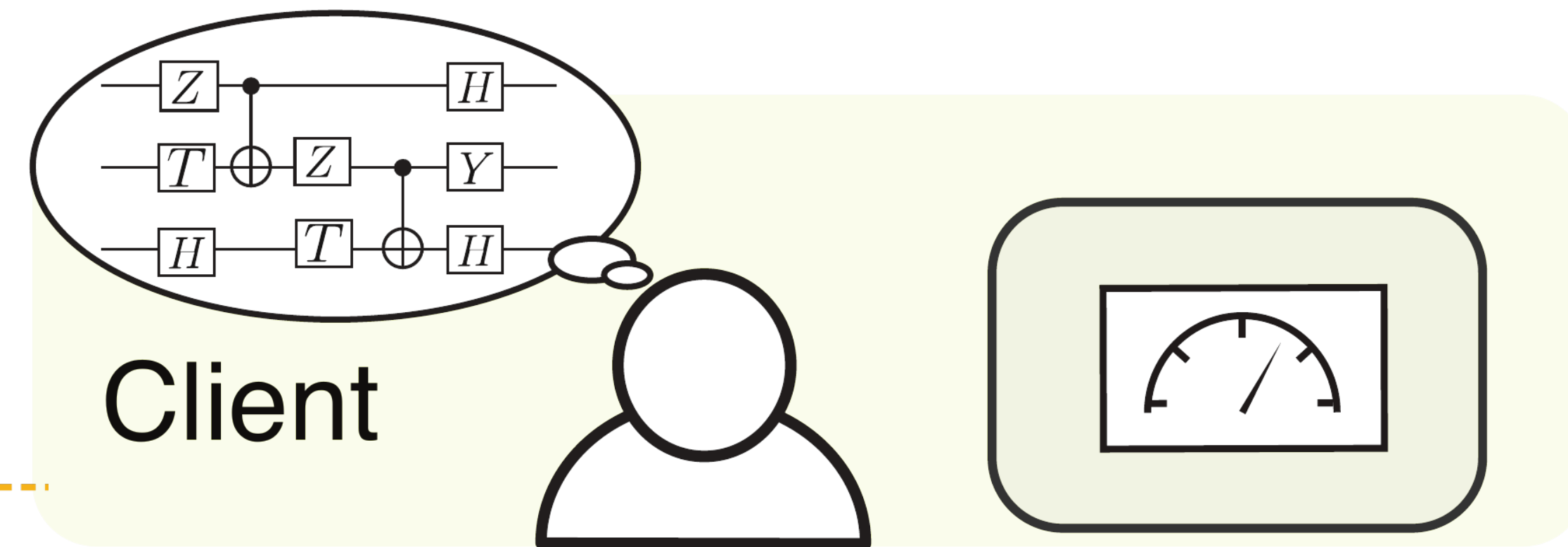
Server:

Matter-Based Fault-Tolerant
Quantum Computer



Client:

Photonic Measurement Device
+ Classical computer



Classical Information

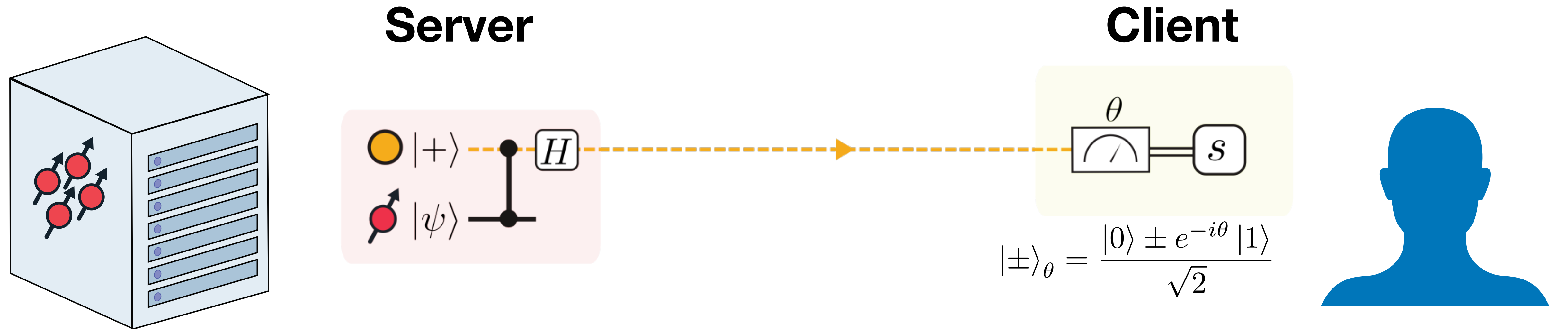


Photonic qubits

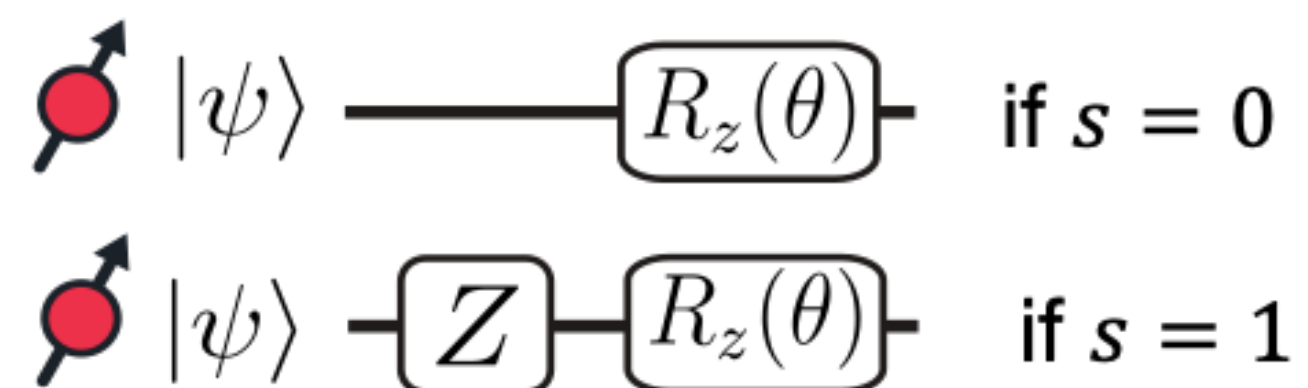


Client's goal: Perform a large quantum computation on
the server without revealing details of the computation

Building a blind rotation gate



After successful measurement:



$$U_s = \boxed{Z^s} R_z(\theta)$$

Pauli Frame

Server perspective:

Random measurements $\sim$ dephasing
Hides angle information

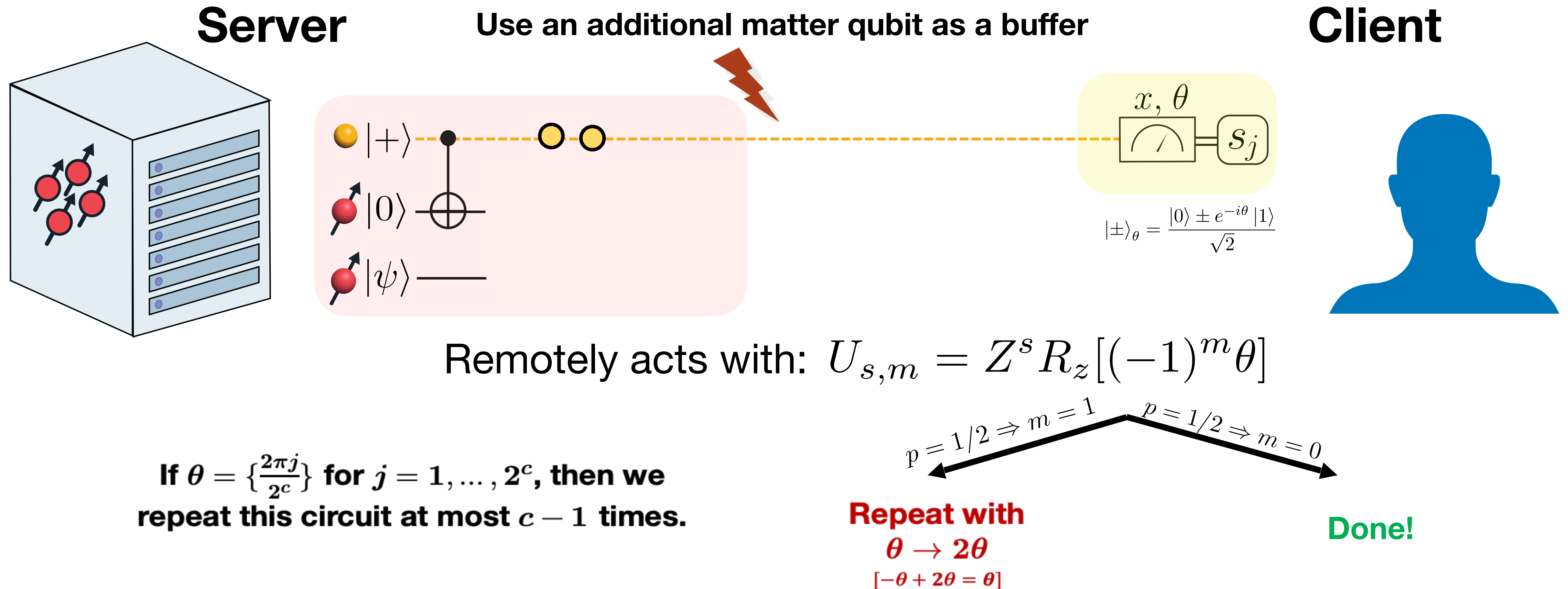
Not yet efficient:
 $P(\text{success}) \sim \eta^{-N}$

Probability of photon arriving η

Client perspective:

Measurement record enables decoding

Building a deterministic blind rotation gate



Never need to abort entire computation $\eta^{-N} \Rightarrow 2\eta^{-1} N$

Why it works: The Pauli Frame

Applying a blind gate:

Pauli Frame

$$Z^s R_z(\theta) X^a Z^b |\psi\rangle = X^a Z^{s+b} R_z[(-1)^a \theta] |\psi\rangle$$

The client knows the measurements = can adapt angles to perform correct computation

Only requirement: gates must preserve the structure of the Pauli frame

Server can apply any Clifford gate **C** non-blindly

$$C P U |\psi\rangle = P' C U |\psi\rangle$$

- 1) Preserves the structure of the Pauli frame

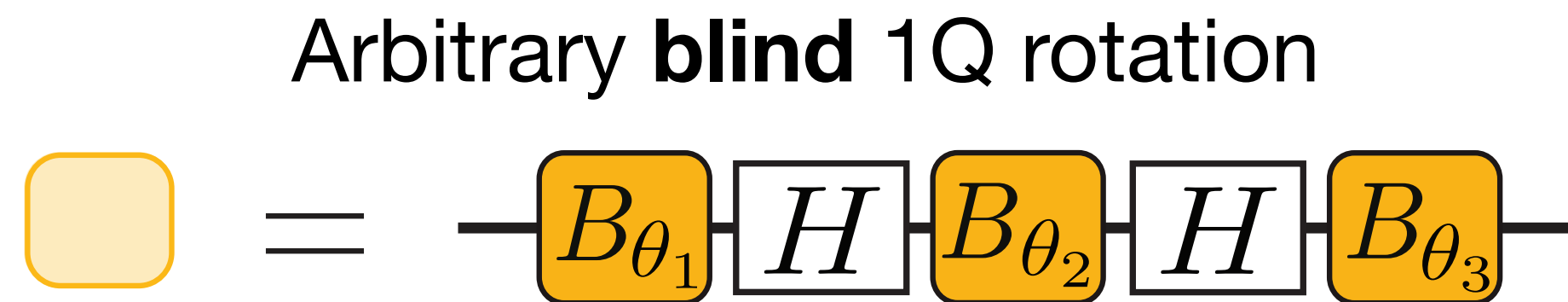
$$C^\dagger P C = P'$$

- 2) The transformation of the Pauli $P \rightarrow P'$ is classically computable so the client can keep track of changing frame

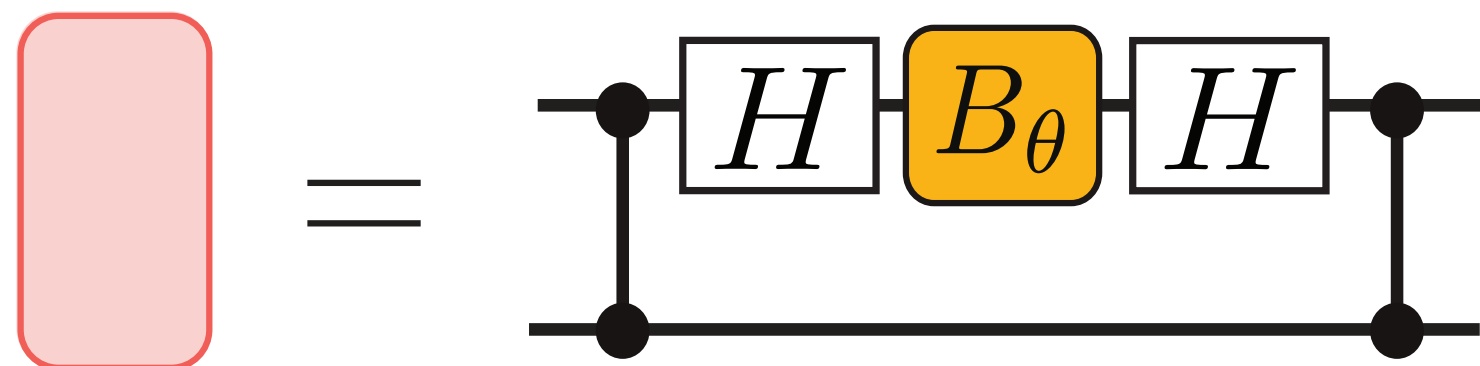
Putting these elements together!

Building more complex gates

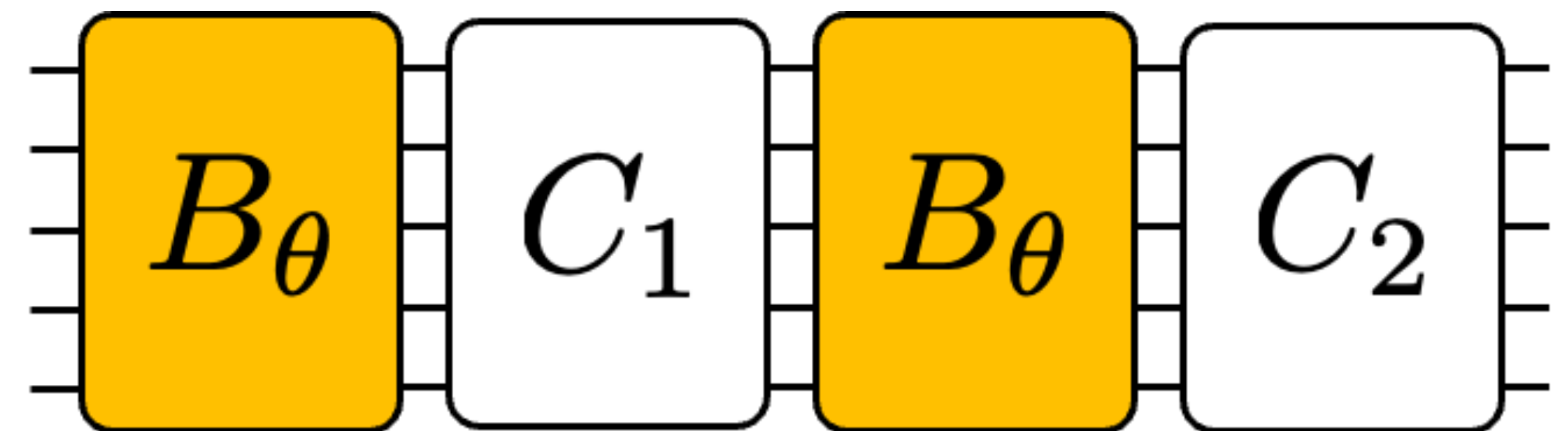
It is a **universal** primitive



Entangling 2Q rotation



It is a **flexible** primitive



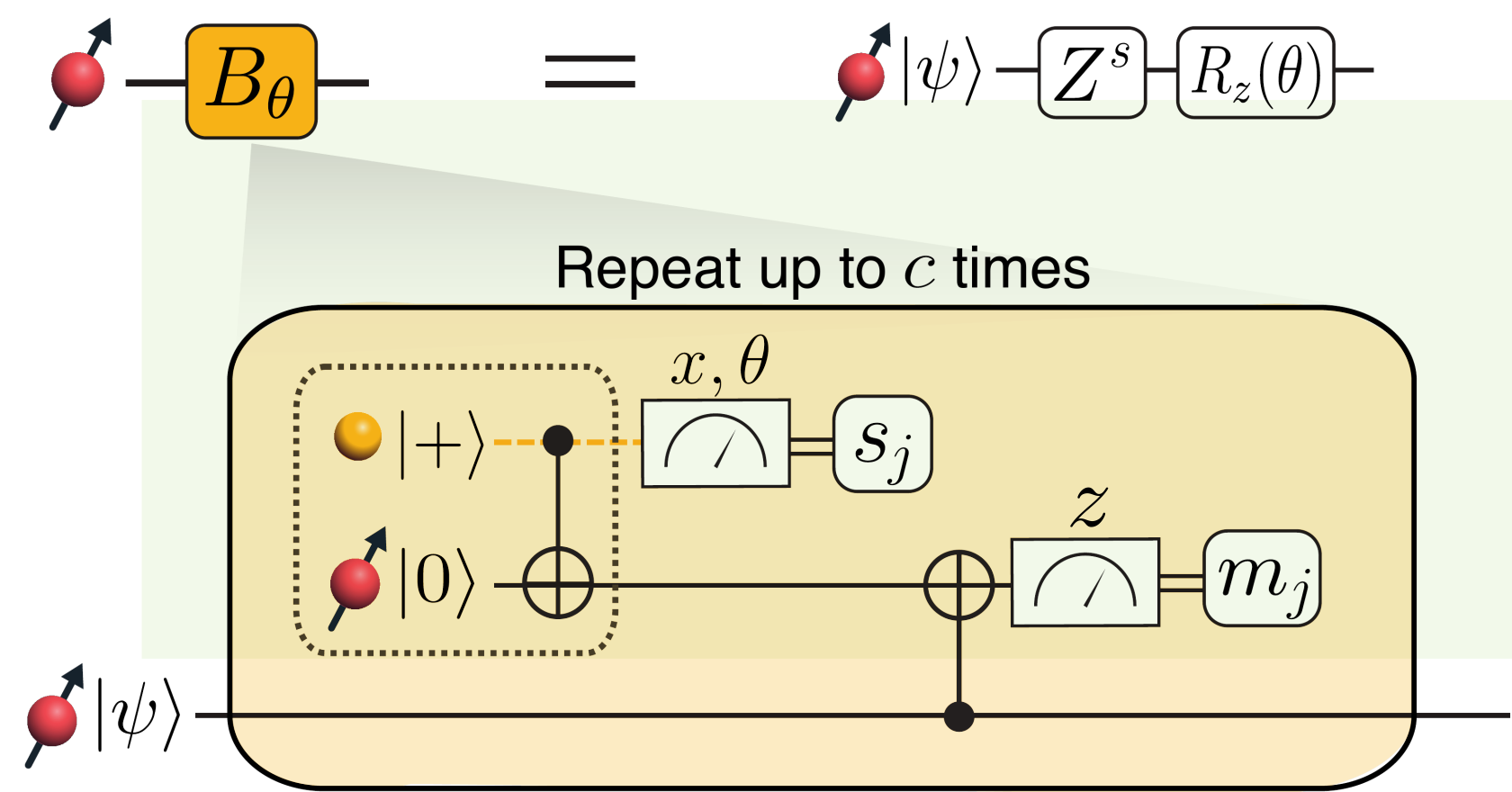
Interlacing **blind** and **known** gates:

- 1) Reduces communication costs
- 2) Client can **choose which information** they hide

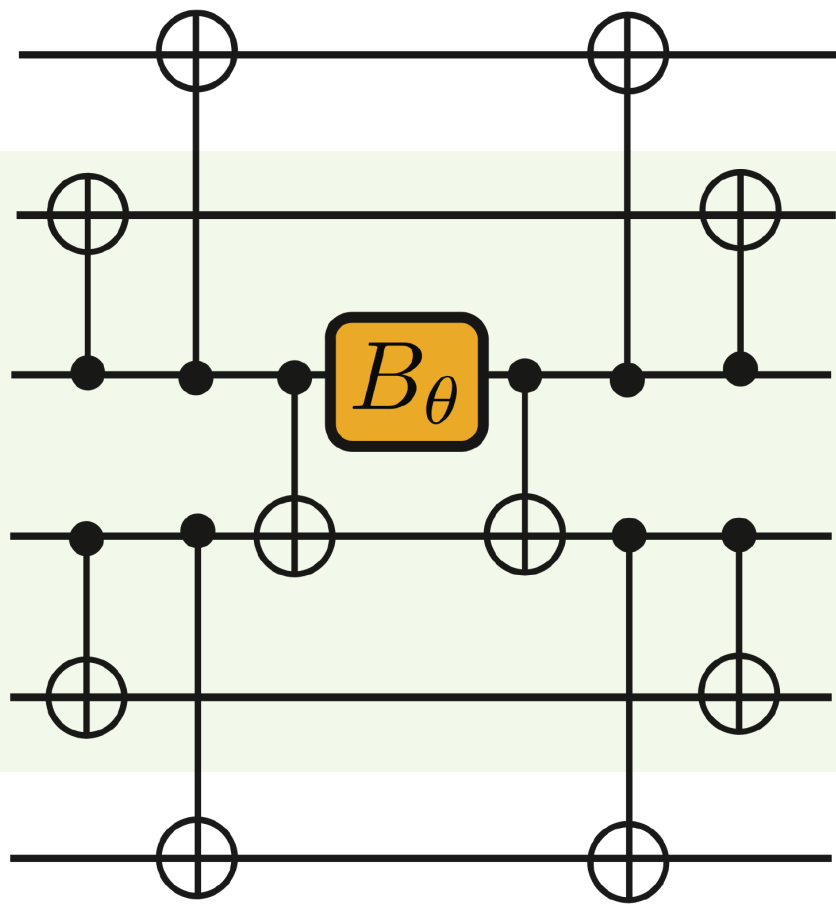
Enough to do everything (in principle). **What can we do in practice?**

Designing Fault-Tolerant Blind Quantum Computing

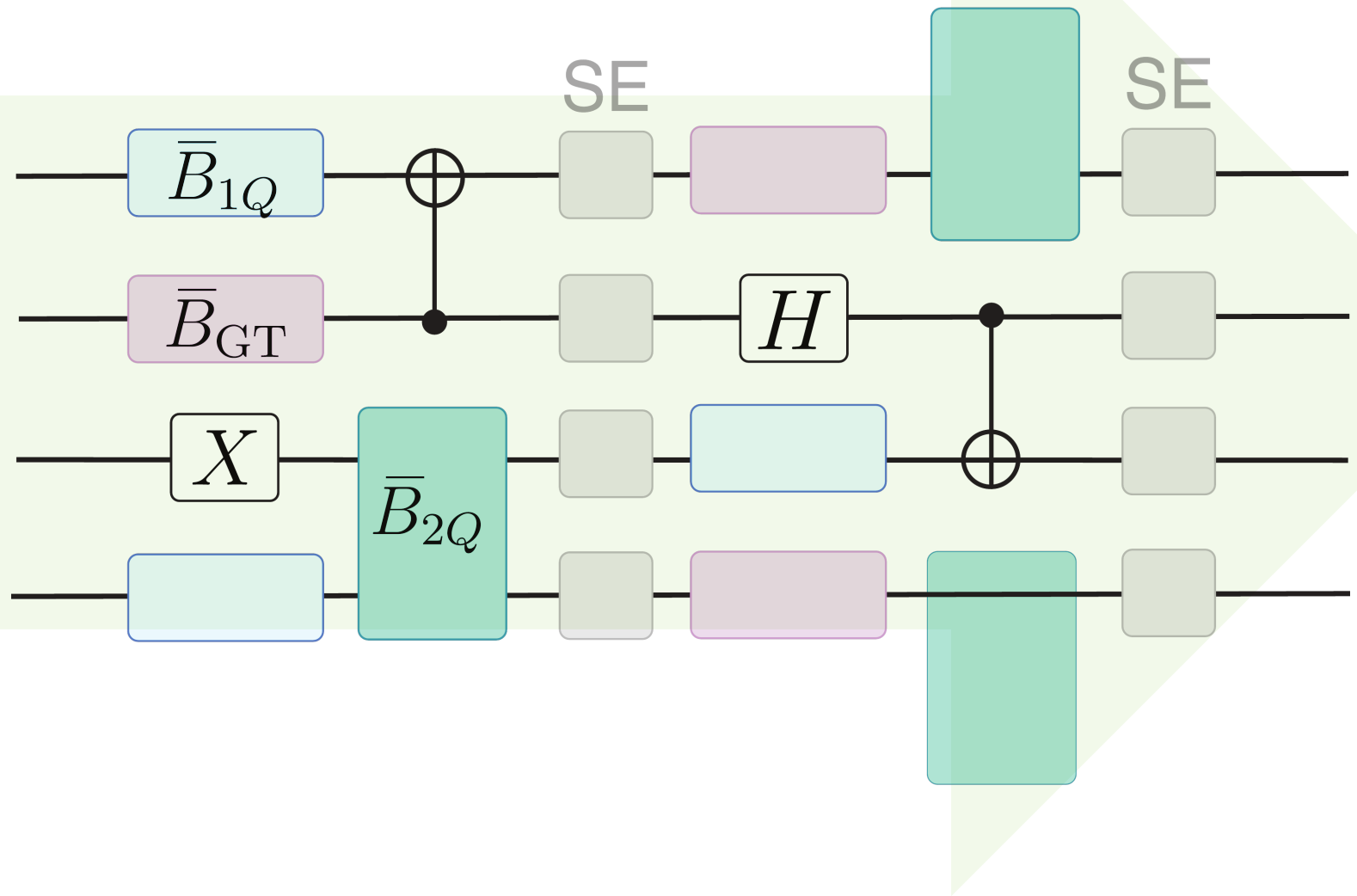
Hybrid Light-Matter Blind QC



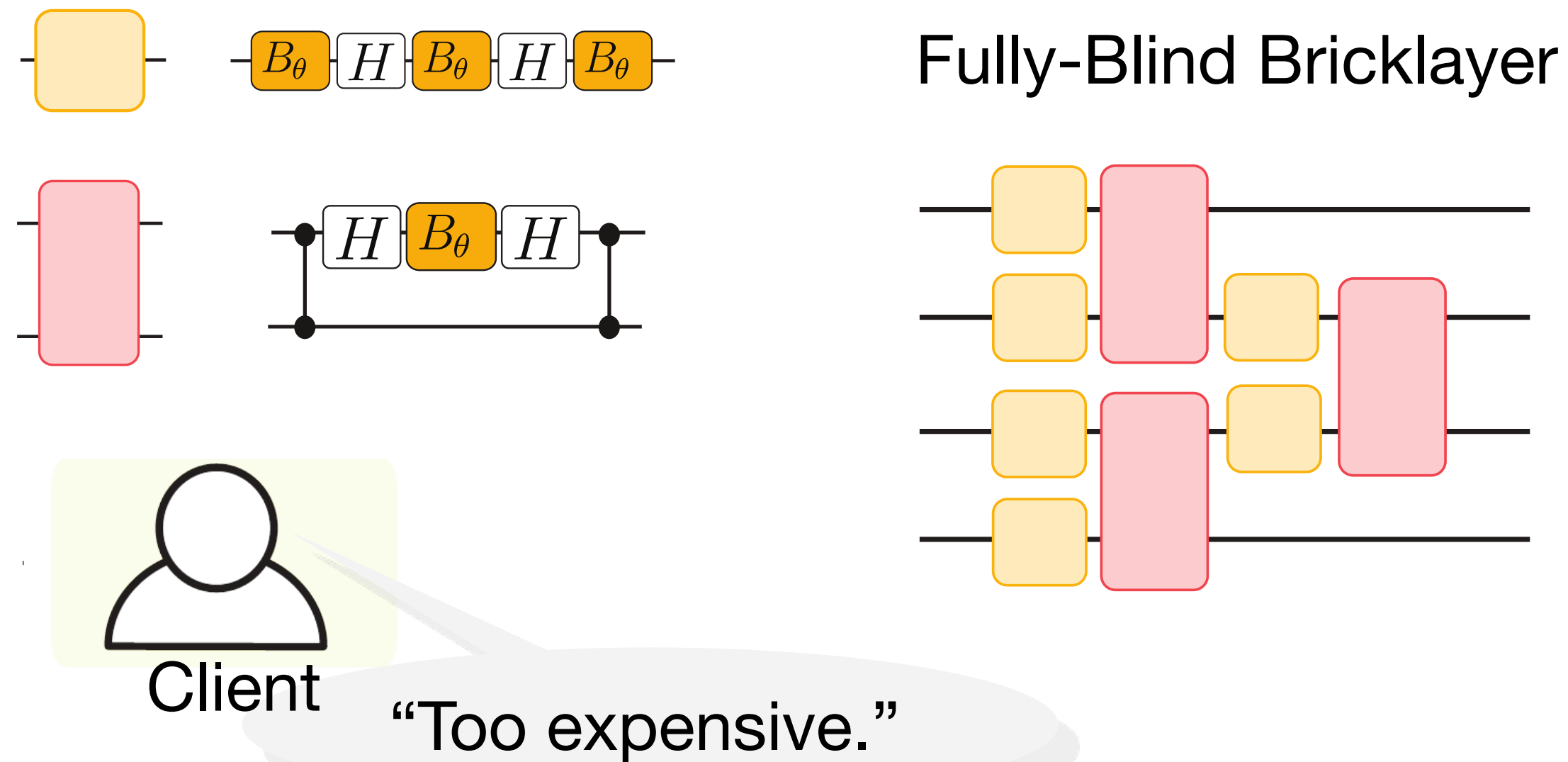
Exploring new circuits design possibilities



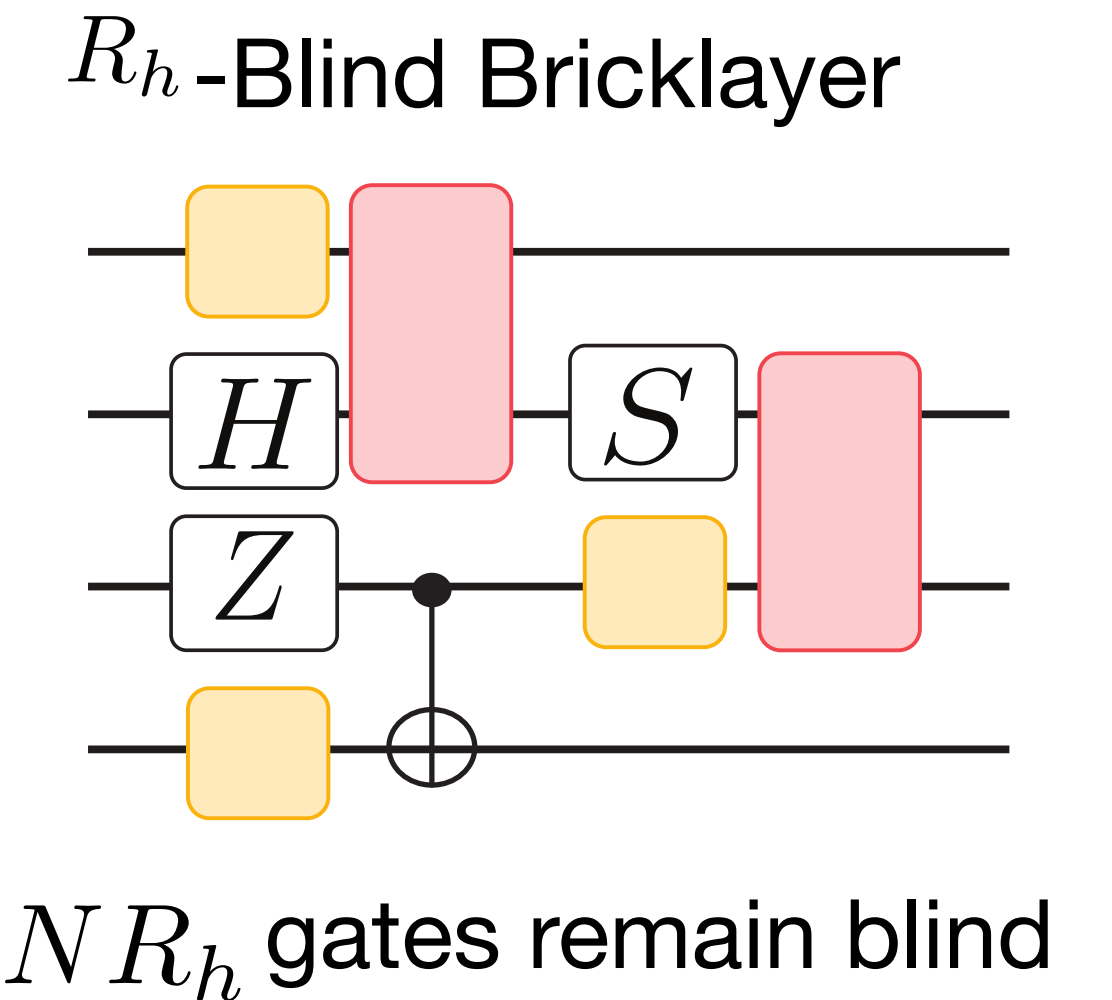
Fault-tolerant scalable implementation



Scenario 1: Blindness-Efficiency Trade-Off



Make some gates known to the server



Efficiency: $O\left(\frac{NR_h}{\eta}\right)$

Fidelity: $O(f^{NR_h})$

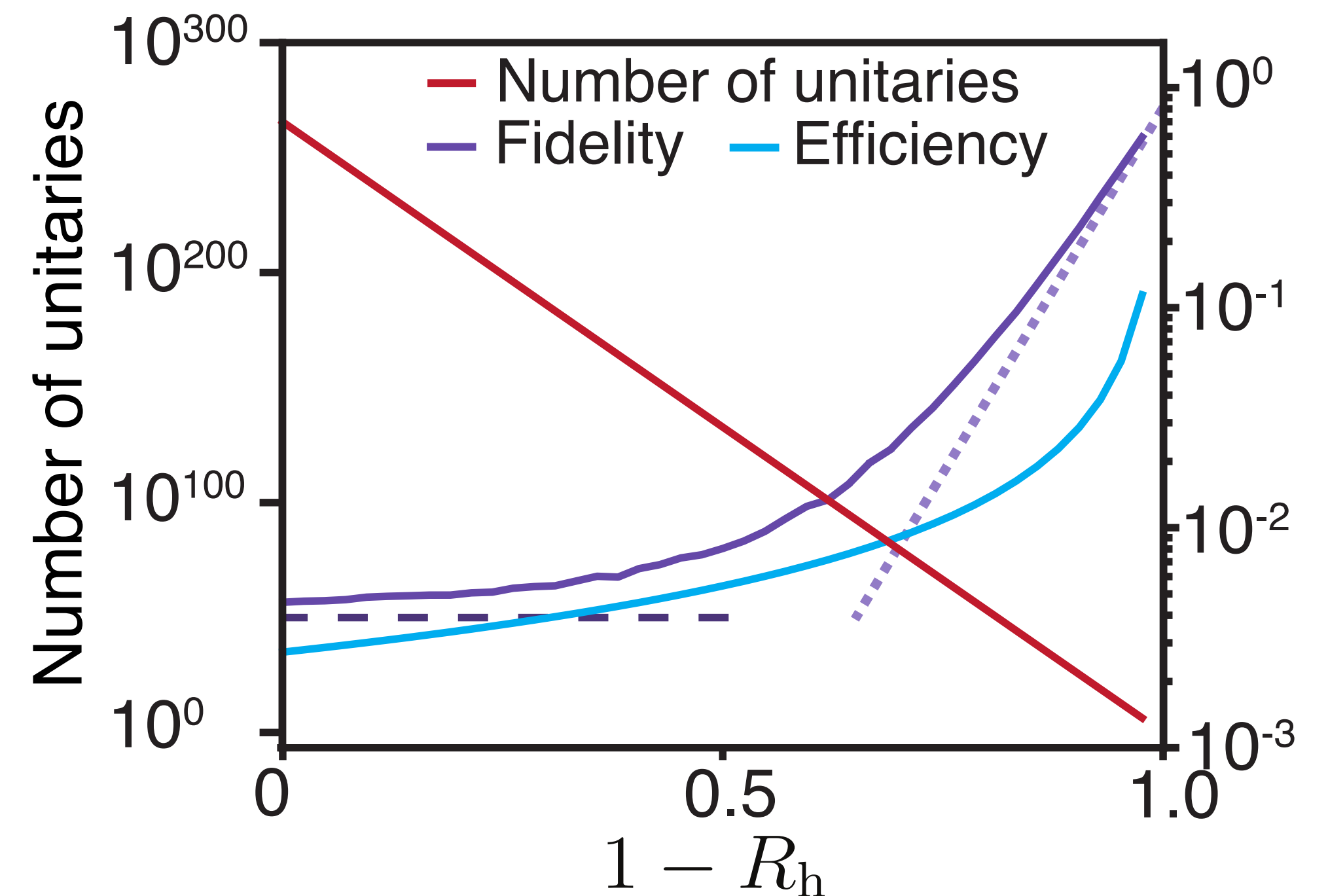
vs

f : blind gate fidelity

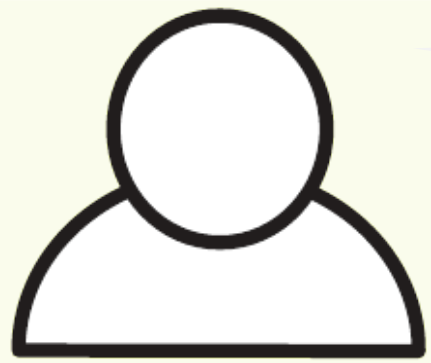
Number of Unitaries: $O(2^{cNR_h})$

[A measure of how likely the server is to guess the unitary]

Open question: Is it the **right** measure?



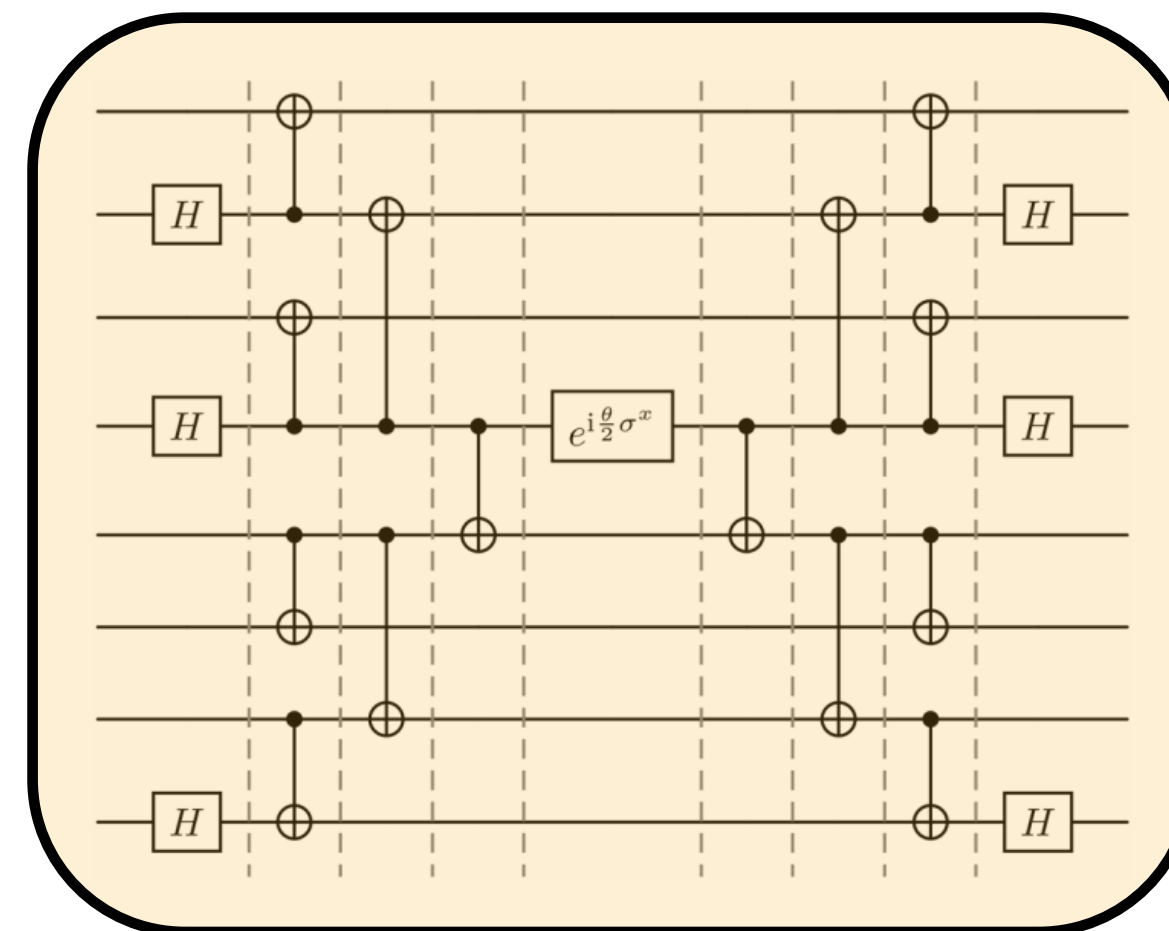
Scenario 2: Hiding Specific Information (Partial Blindness)



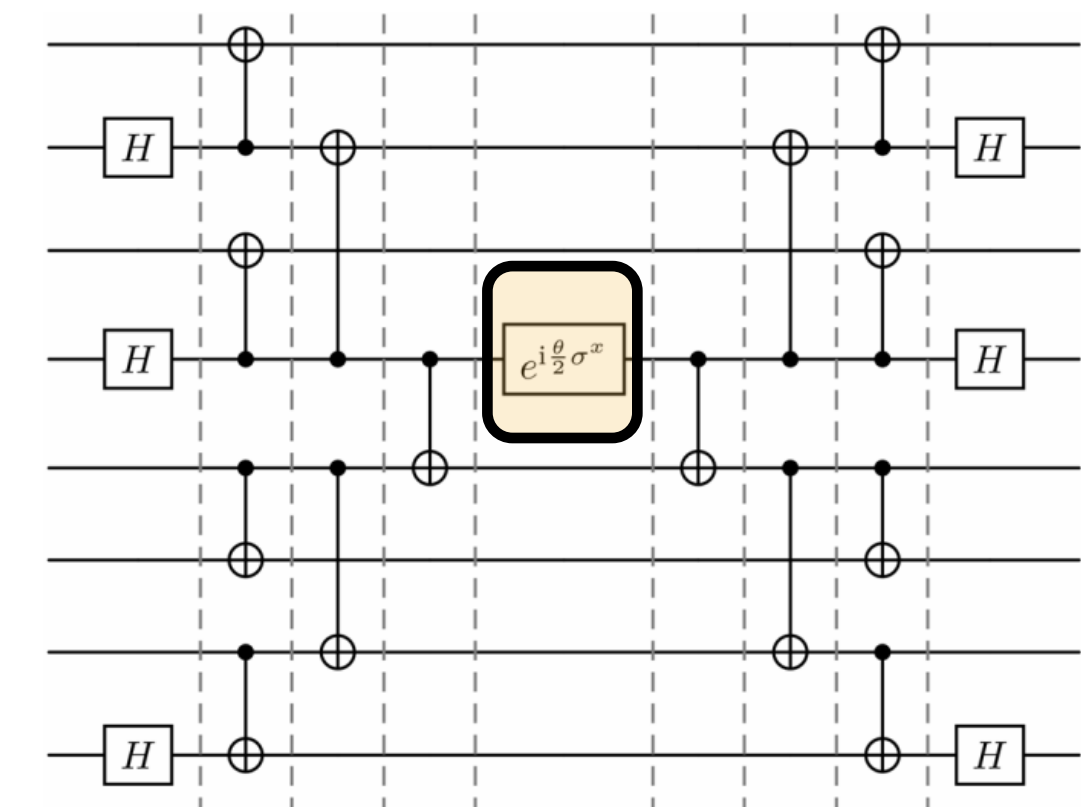
Client

“I do not mind revealing that I am performing Hamiltonian simulation.
I just don’t want the server to know what I’m simulating.”

Hiding P and θ^*
 $O(\text{poly}(n))$ blind gates



Hiding θ^*
 $O(1)$ blind gates



$$H = \sum_i h_i P_i$$

Pauli string P_i

Need to implement:

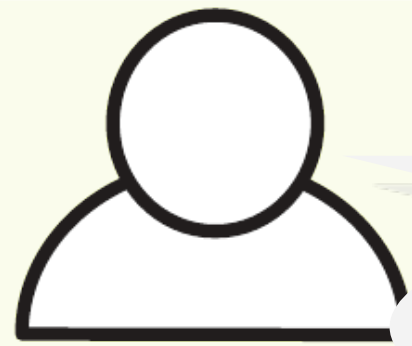
$$e^{-iHt} \approx \left[\prod_i e^{-i(h_i t/K) P_i} \right]^K$$

$$e^{i \frac{\theta_i}{2} P_i} =$$

Key Primitive: $e^{i \frac{\theta_i}{2} P_i}$

The client can choose which aspect of the computation to conceal

Scenario 3: Increasing the Space of Accessible Unitaries



Client

“I can only perform N blind gates.
How ‘complex’ can my computation be?”

Expressibility: the ability of a circuit to generate a uniformly distributed ensemble in the space of all possible unitaries

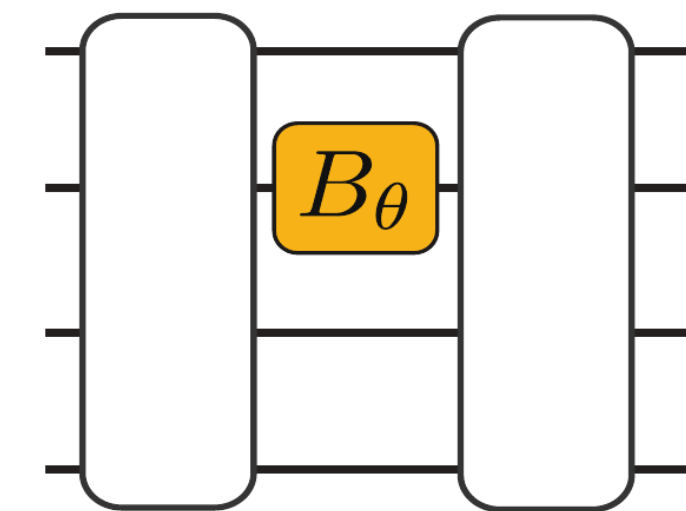
Low expressibility: Blind circuit has access to unitaries with high overlap (i.e. similar)

High expressibility: Circuit has access to a wide variety of distinct unitaries

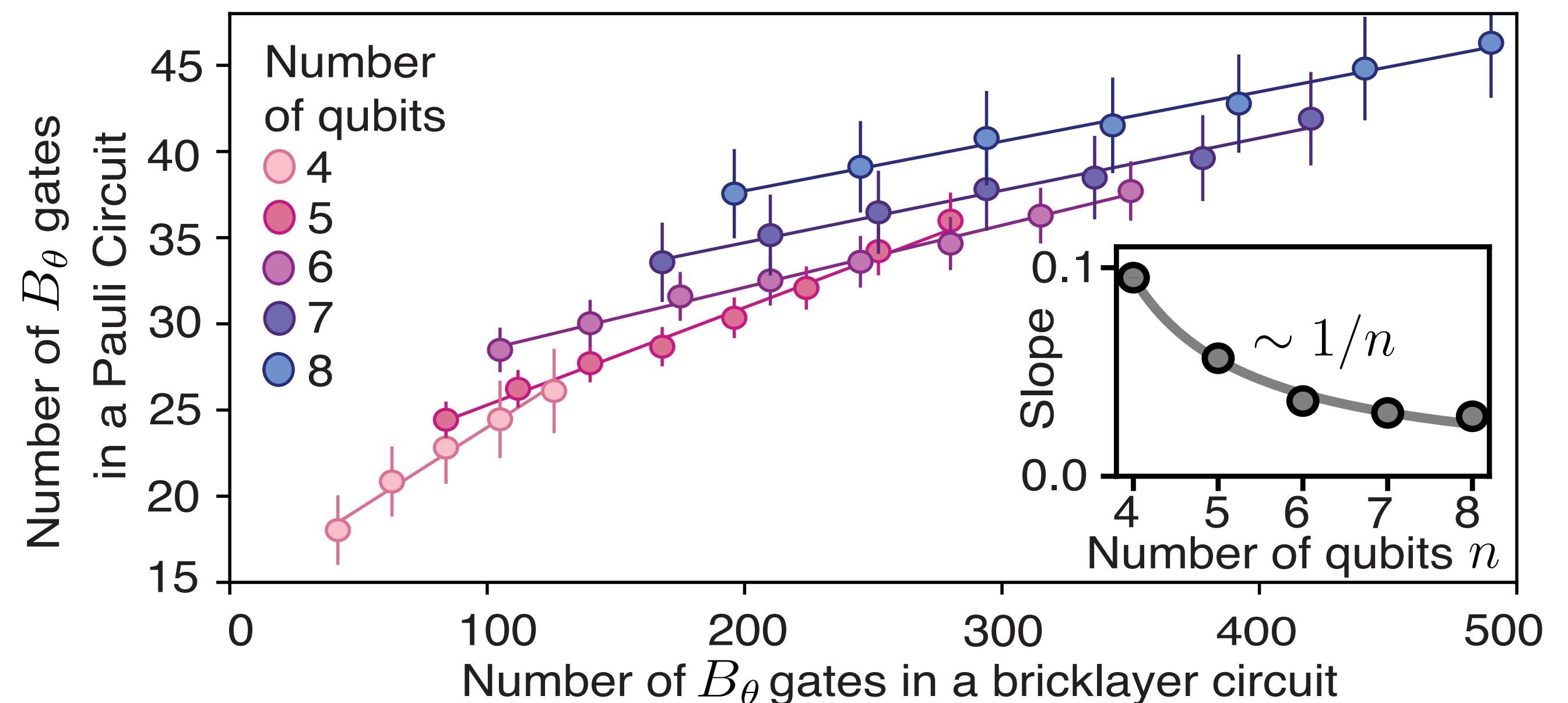
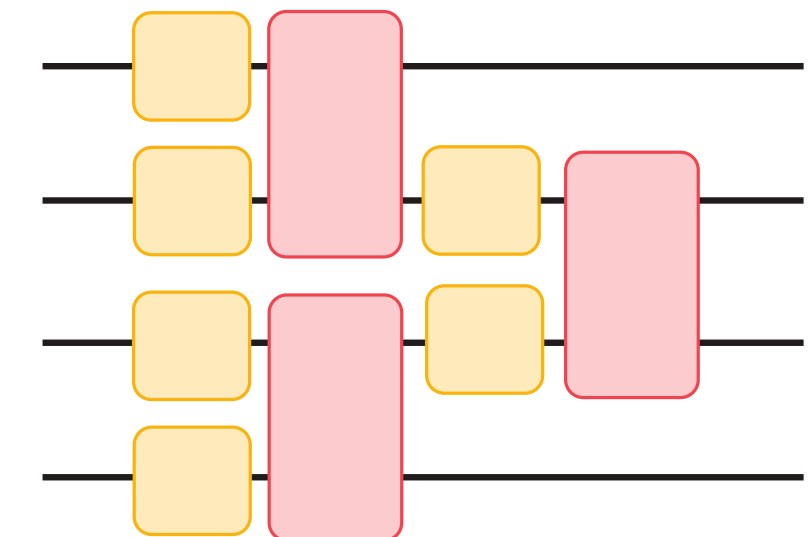
More powerful!

New circuit designs → the space of unitaries accessible to the client expands greatly

Pauli Rotation Circuit



Bricklayer

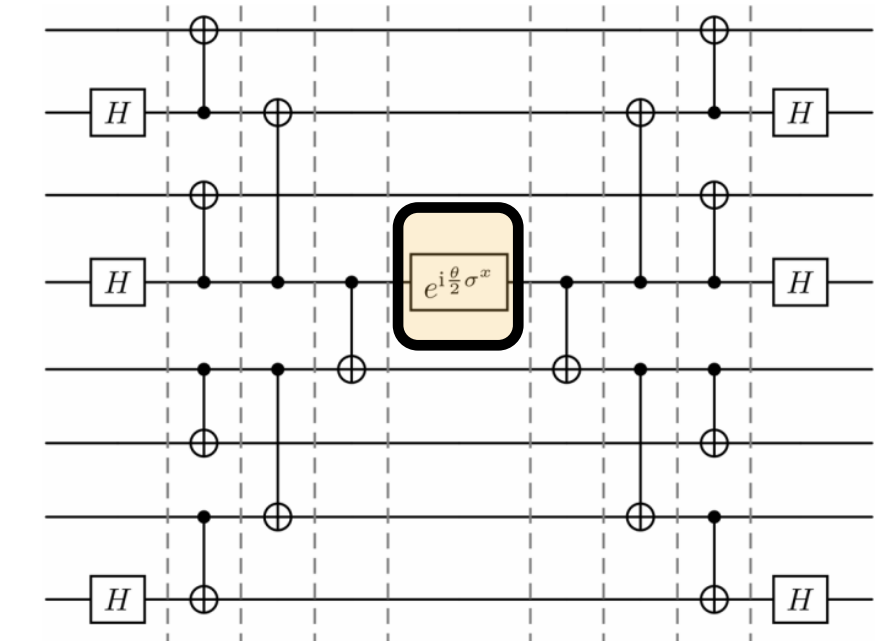
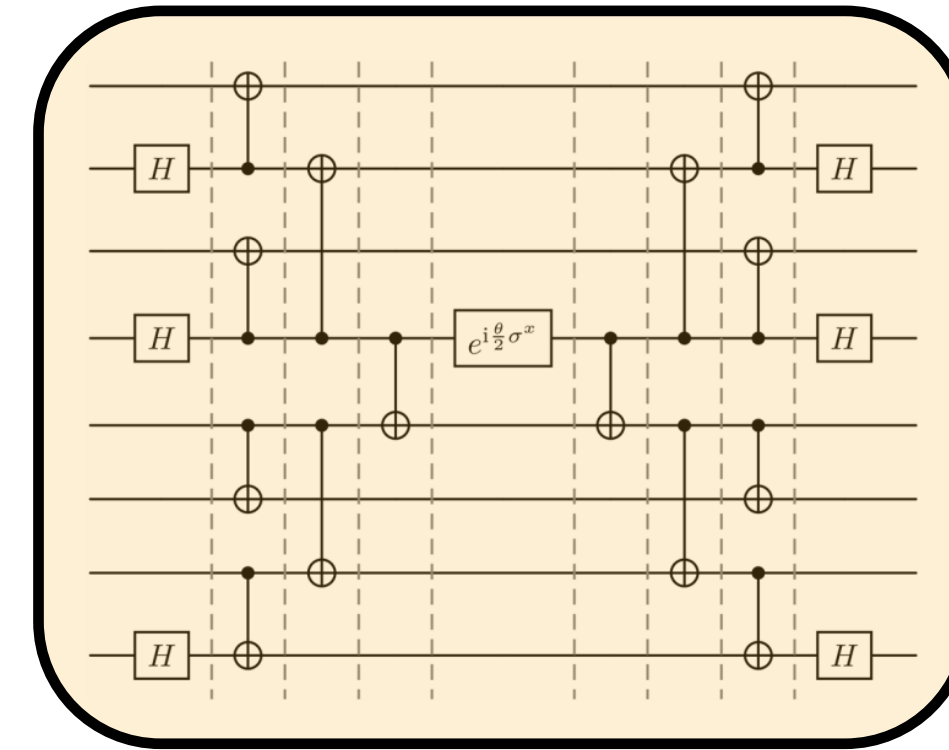


Interplay of Local and Blind gates: 3 Takeaways

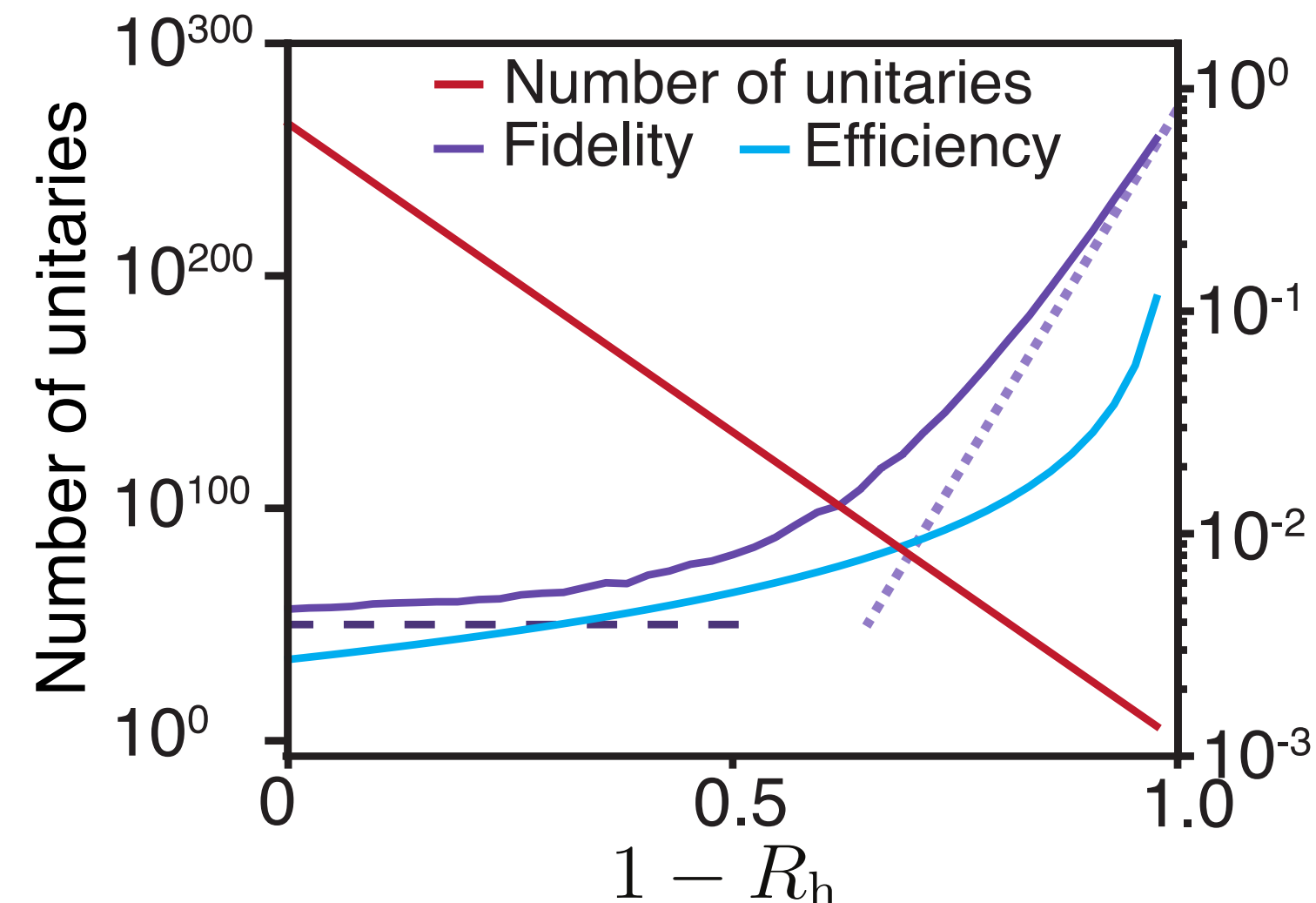
Delegated blind gates
+
Local high-fidelity Clifford unitaries
=
Larger Landscape of Circuit Design Possibilities

More efficient and secure circuit designs

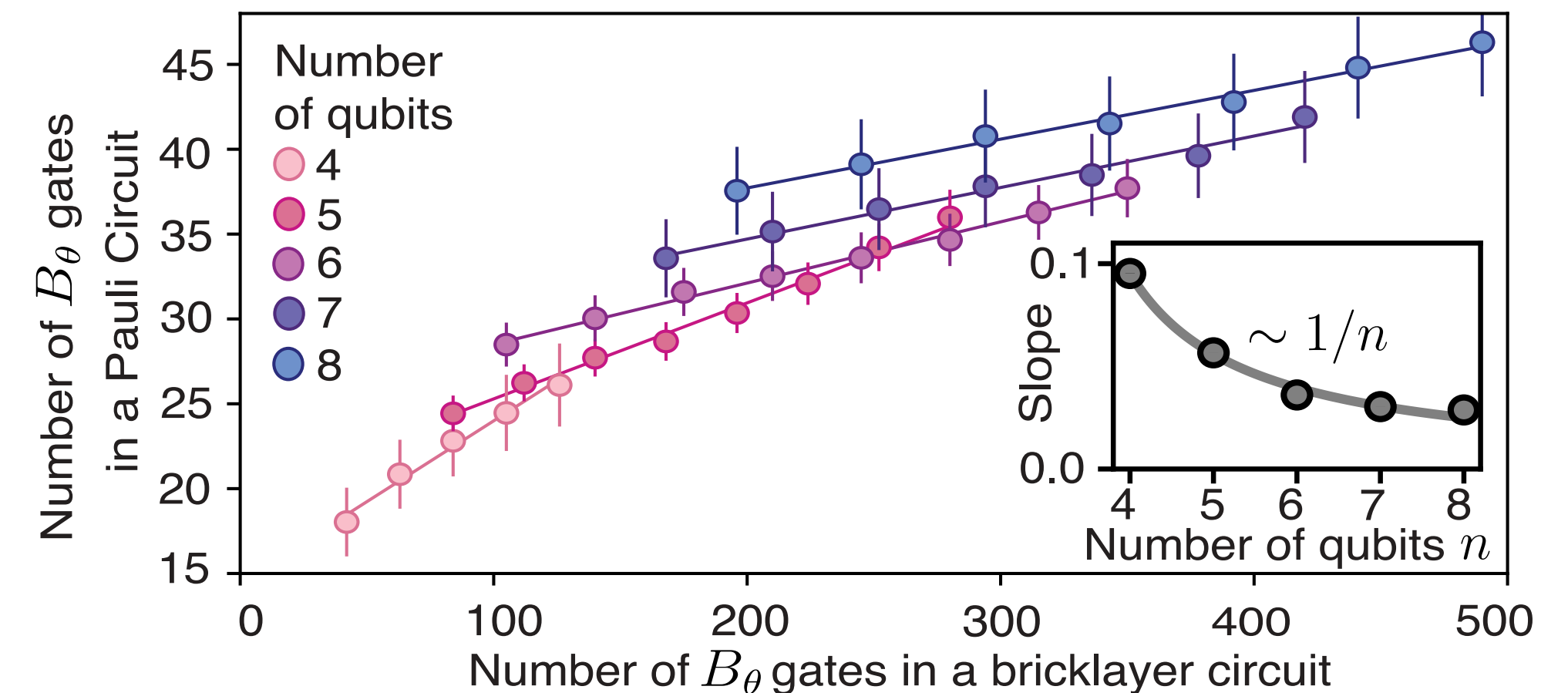
$$e^{i\frac{\theta}{2}P} =$$



Security-Efficiency Trade-Off

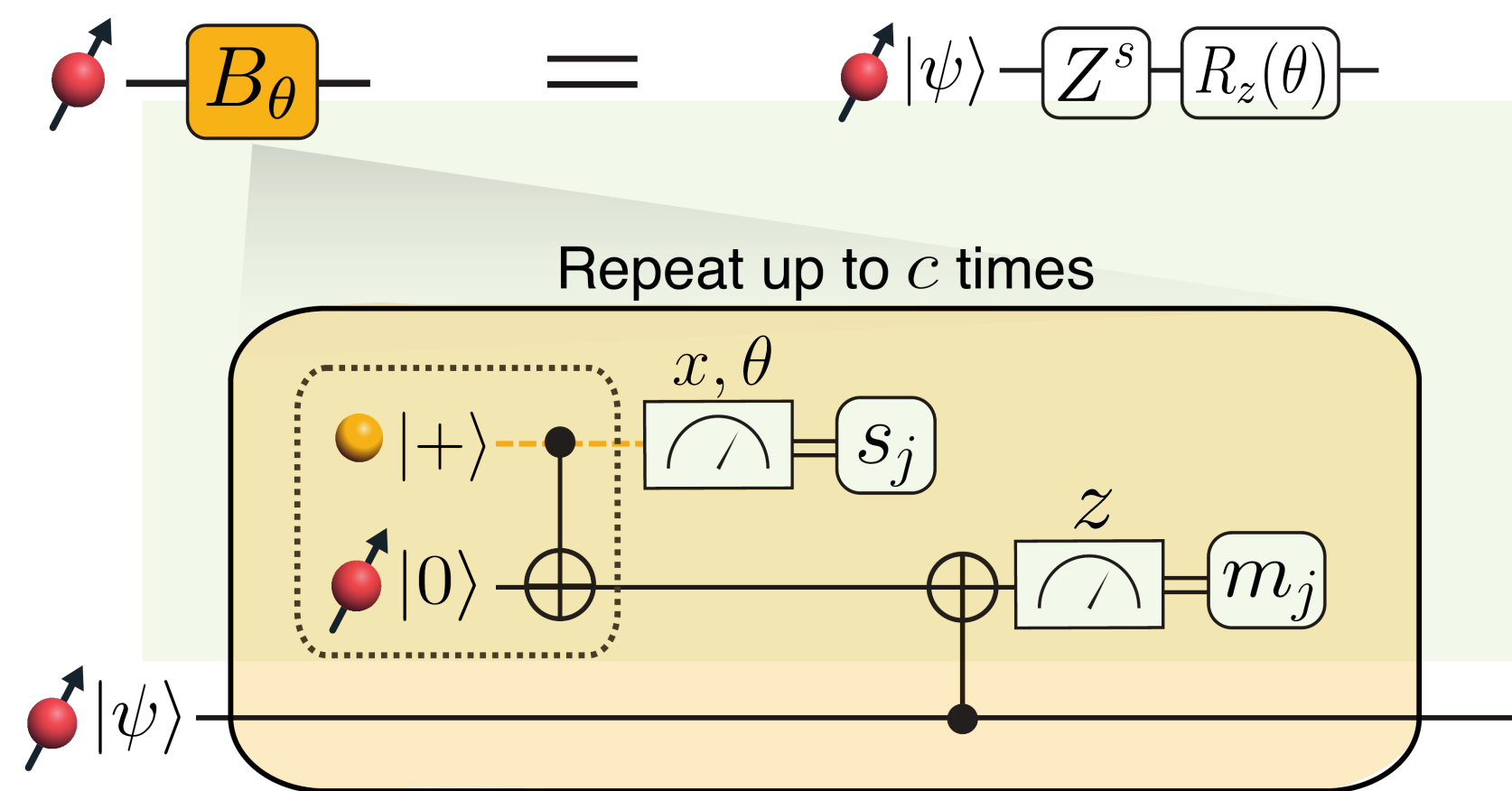


More expressive unitaries with fewer photons

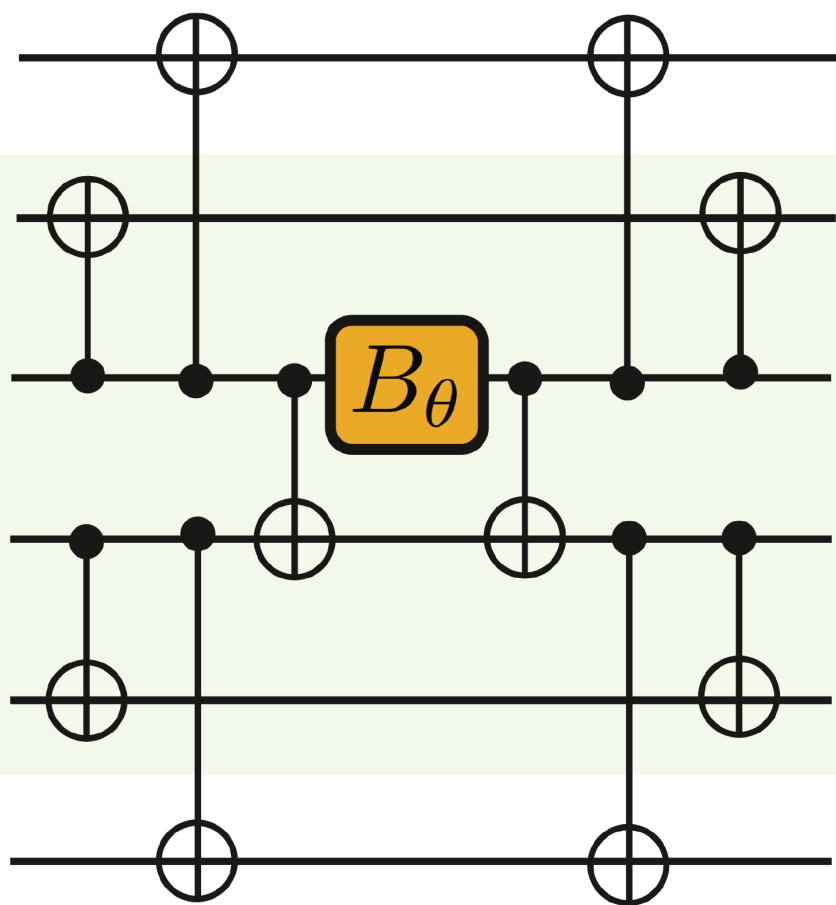


Designing Fault-Tolerant Blind Quantum Computing

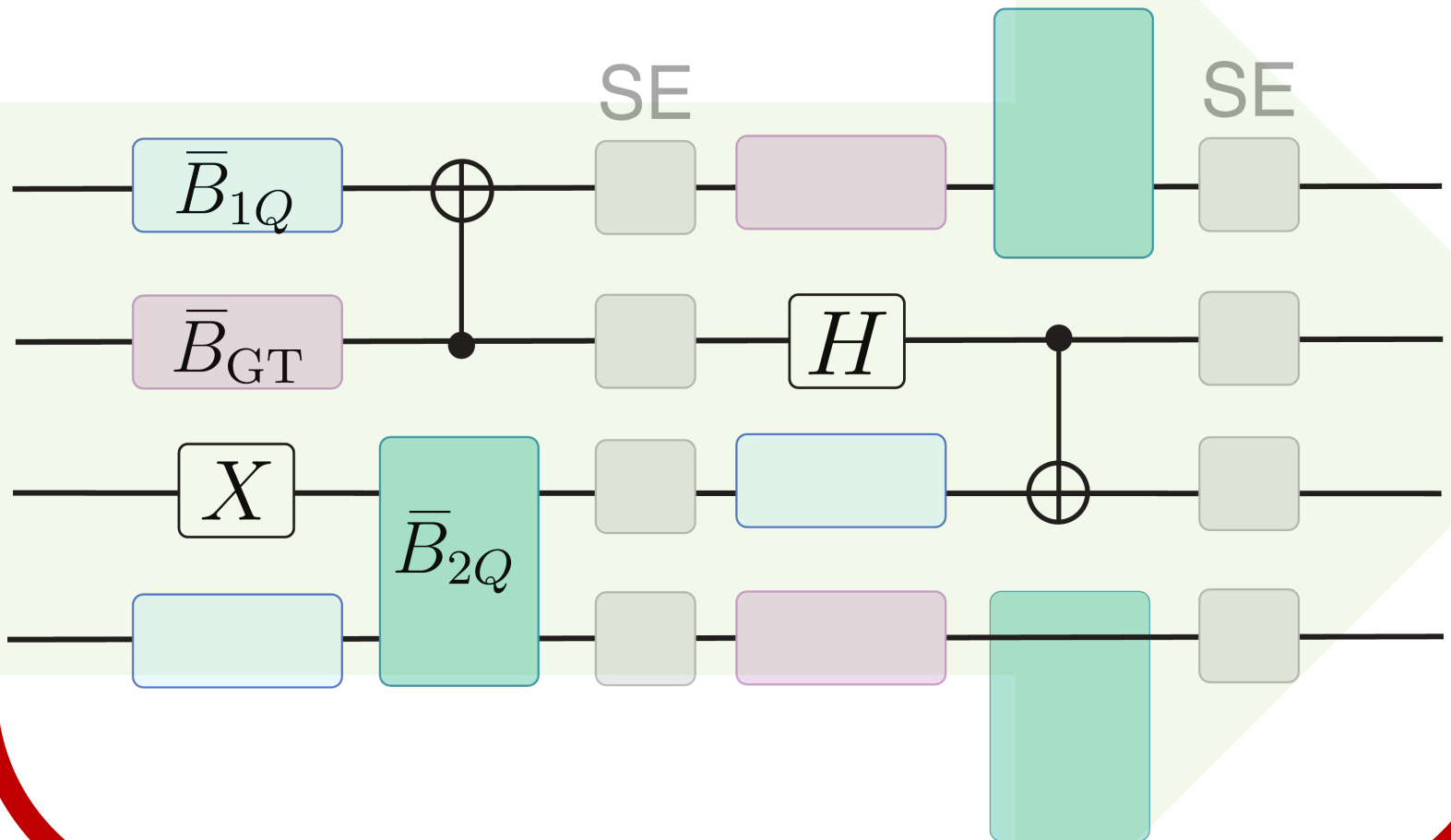
Hybrid Light-Matter Blind QC



Exploring new circuits design possibilities

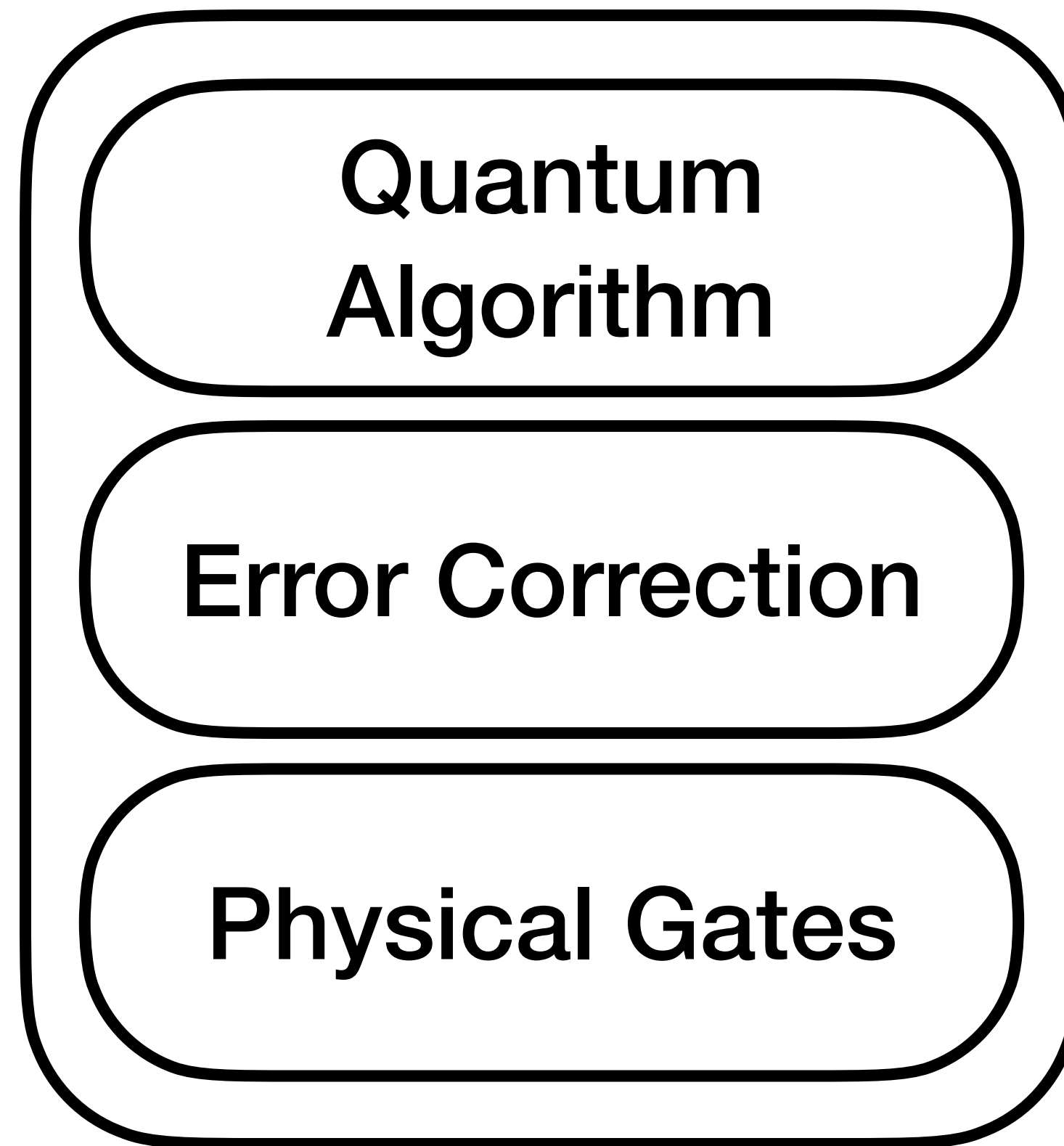


Fault-tolerant scalable implementation

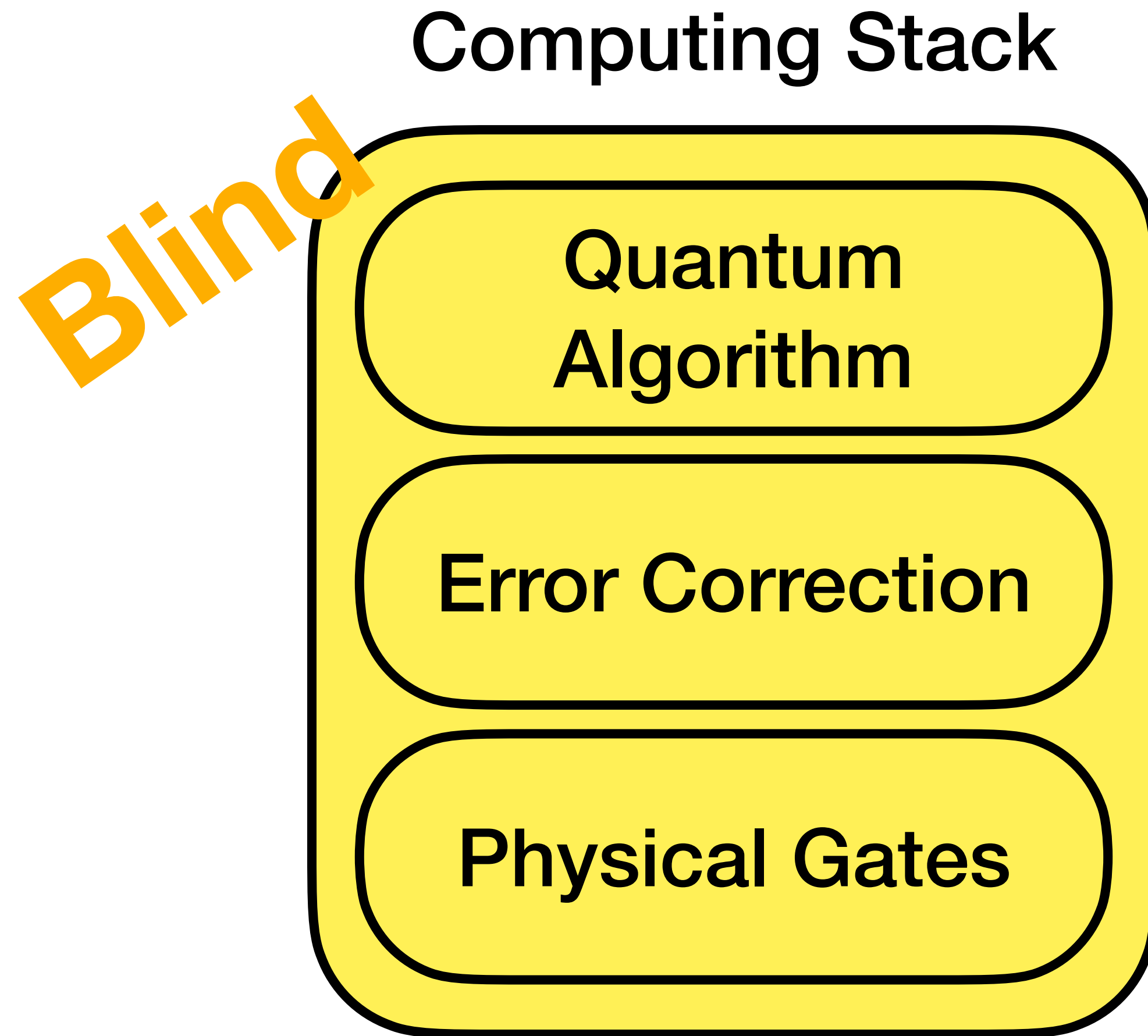


End Goal: Efficient Fault-Tolerant Quantum Computation

Computing Stack

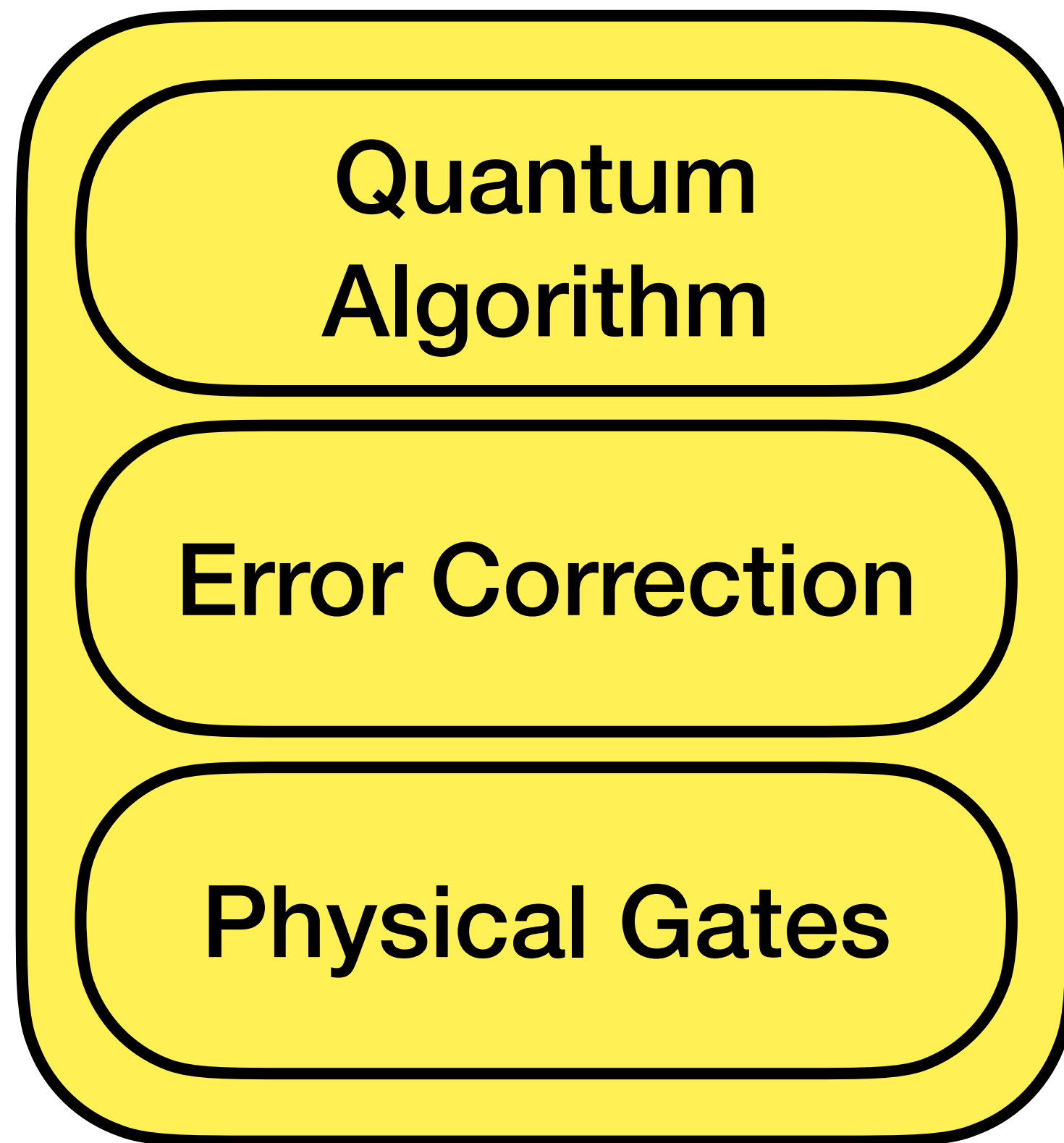


End Goal: Efficient Fault-Tolerant Quantum Computation



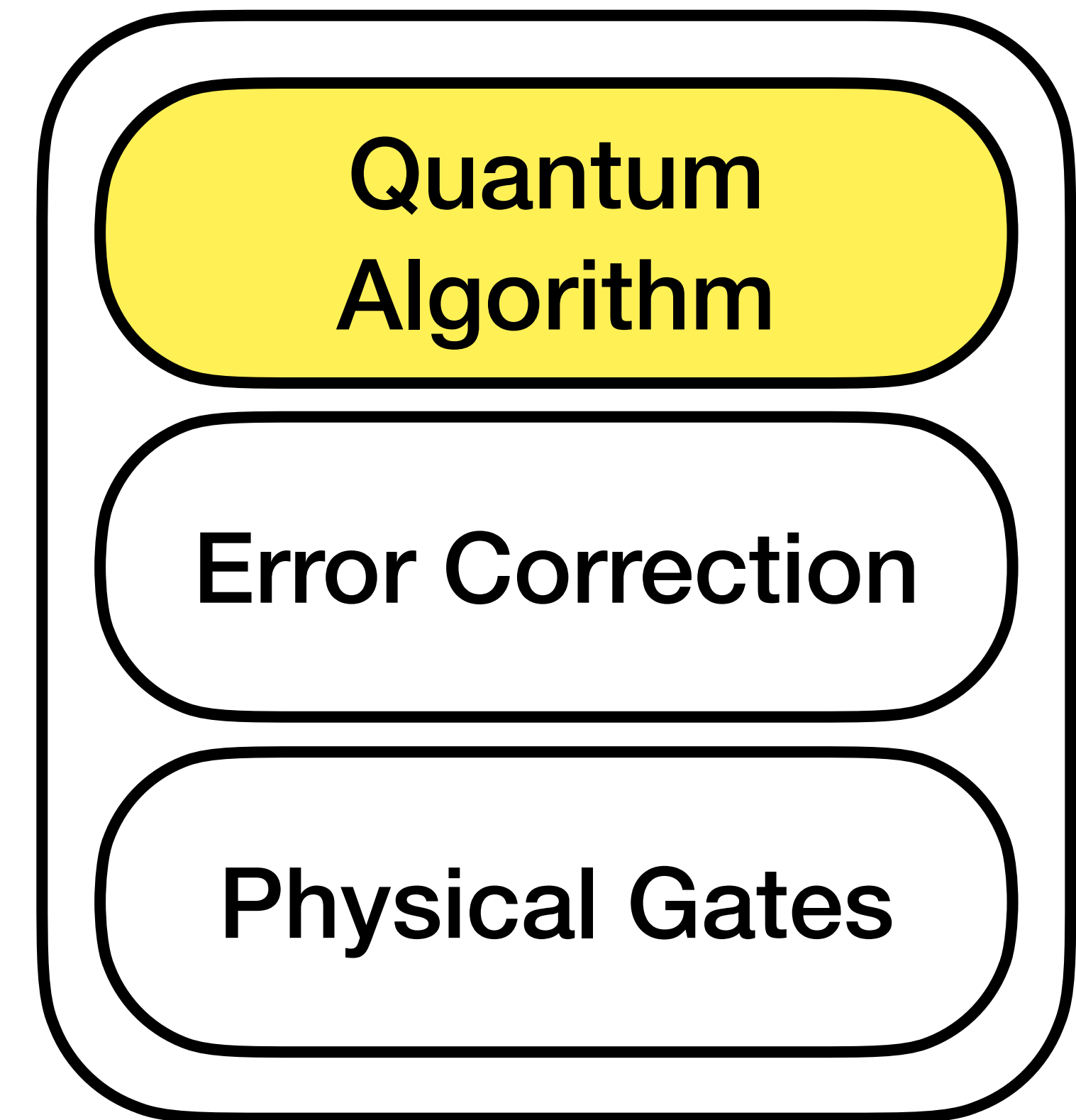
End Goal: Efficient Fault-Tolerant Quantum Computation

Computing Stack

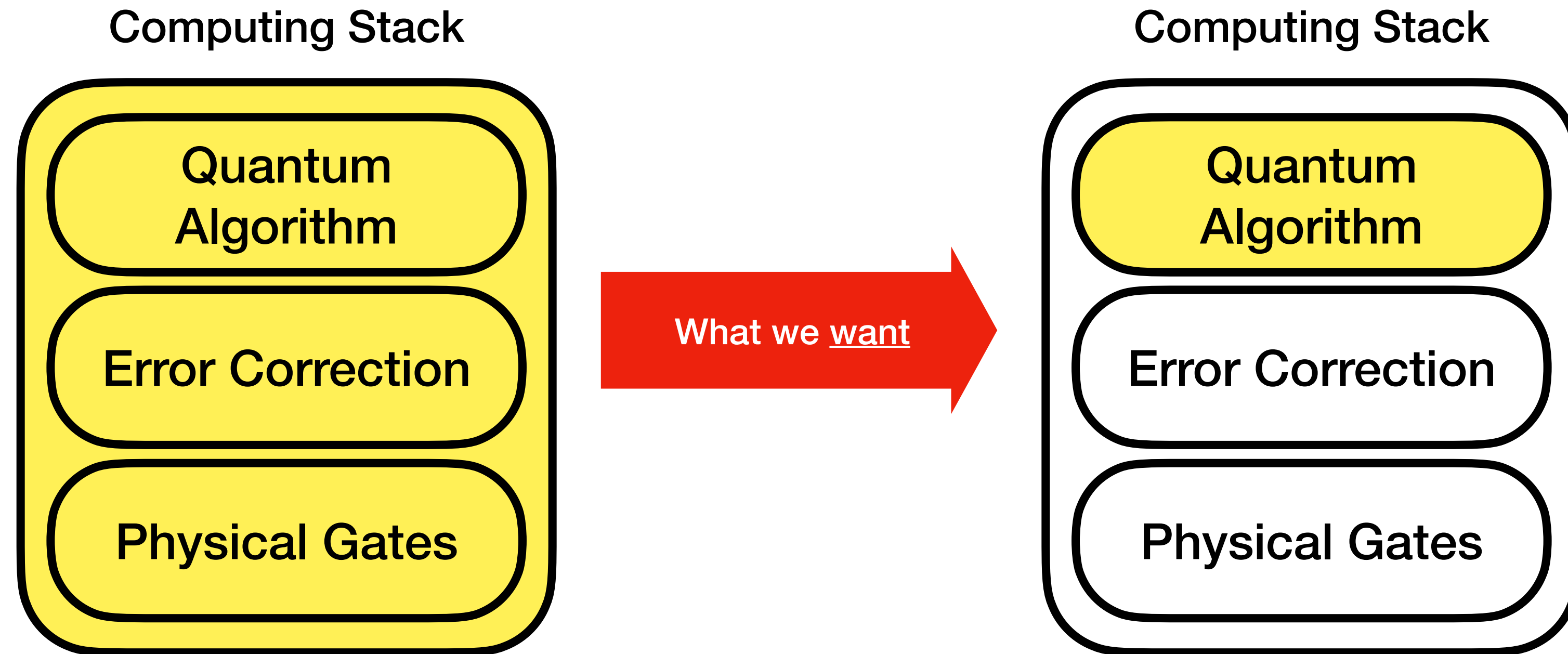


What we want

Computing Stack



End Goal: Efficient Fault-Tolerant Quantum Computation



Question 1:
Is this possible?

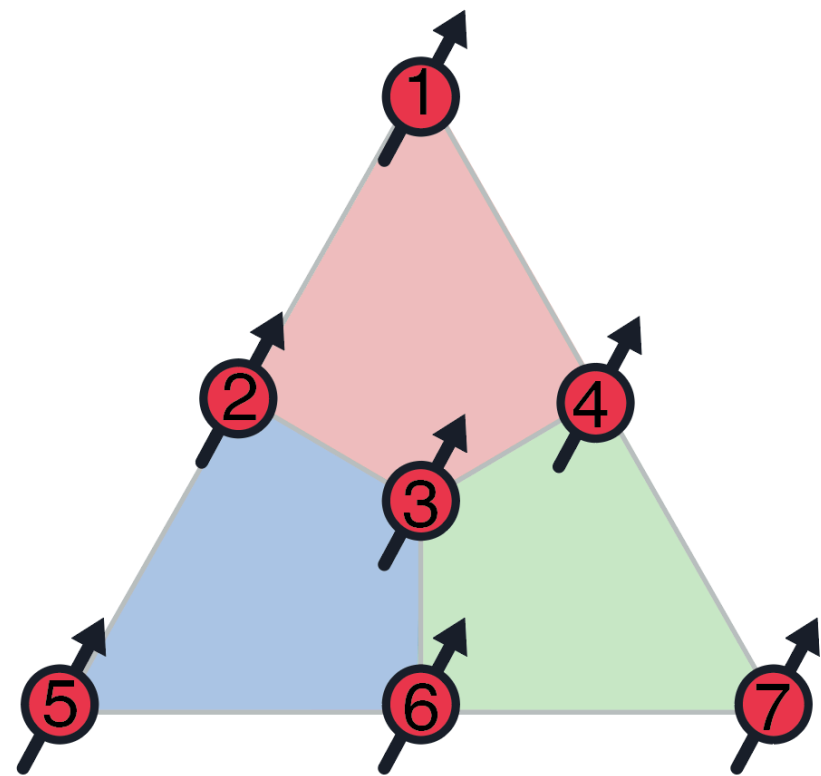
Error correction overhead
cannot reveal information
about the logical operations

Question 2:
Do we have the right architecture?

Matter qubits can form effective
logical qubits, while photons
apply delegated physical gates

General Workflow

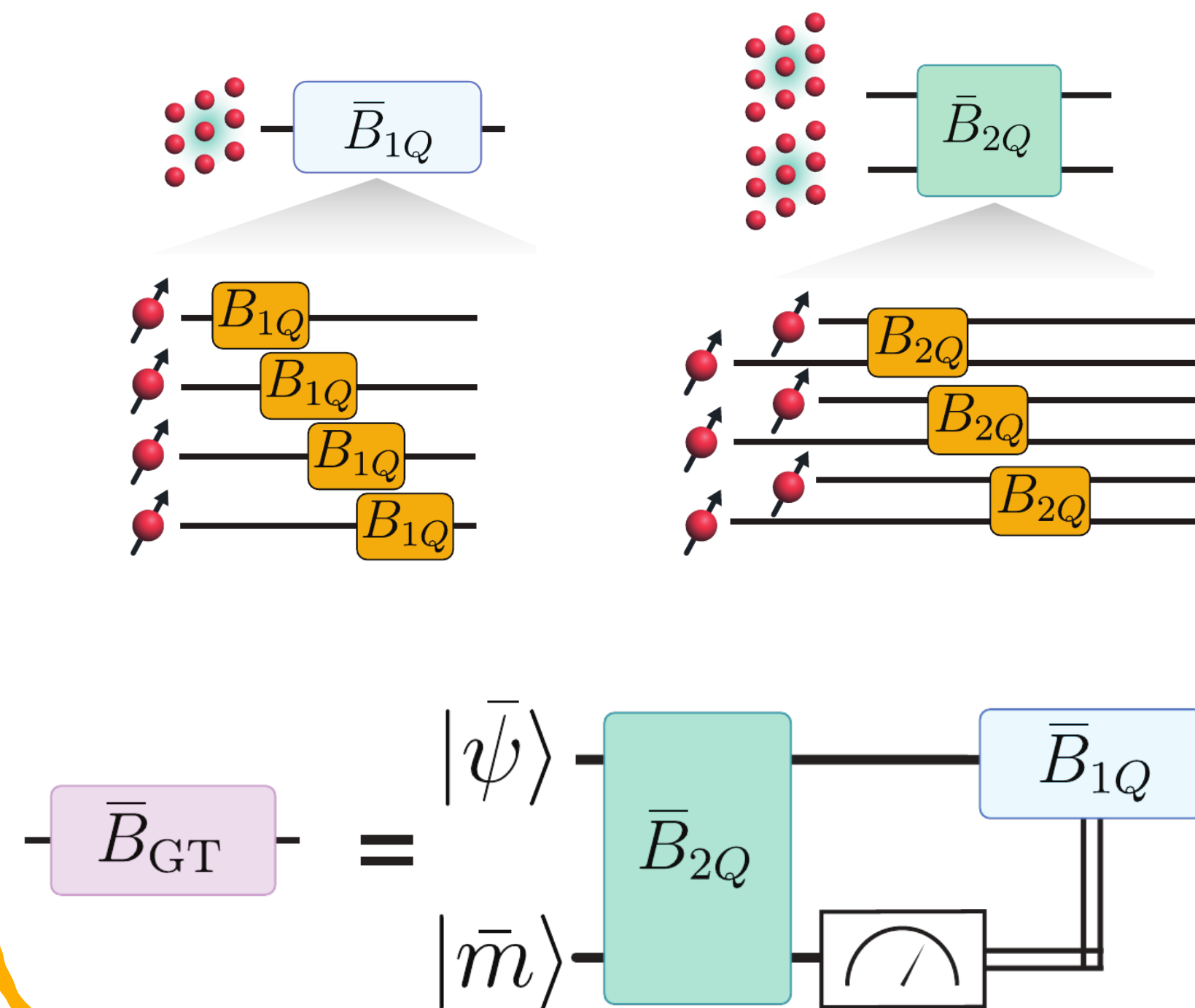
Choose QEC code for the server's qubits



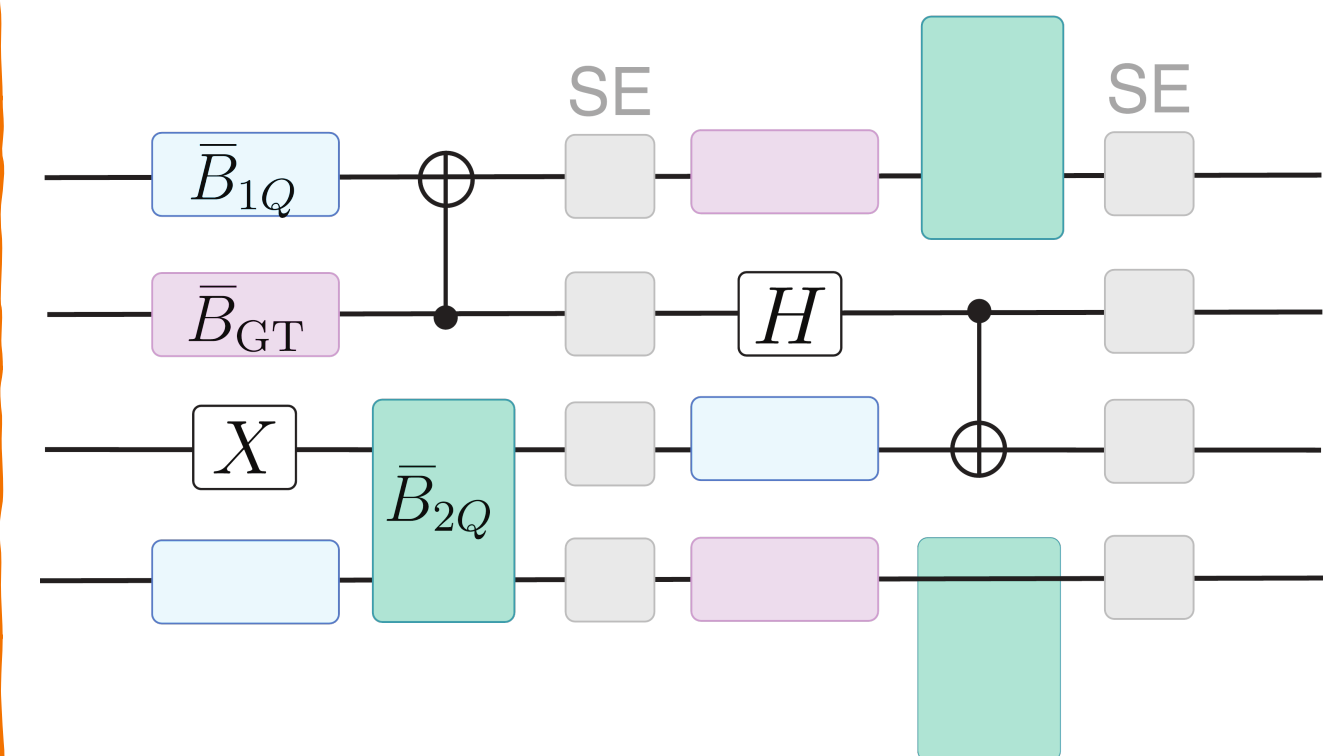
Depending on resources and compilation goals

Determines native gate set

- Transversal
- Teleported



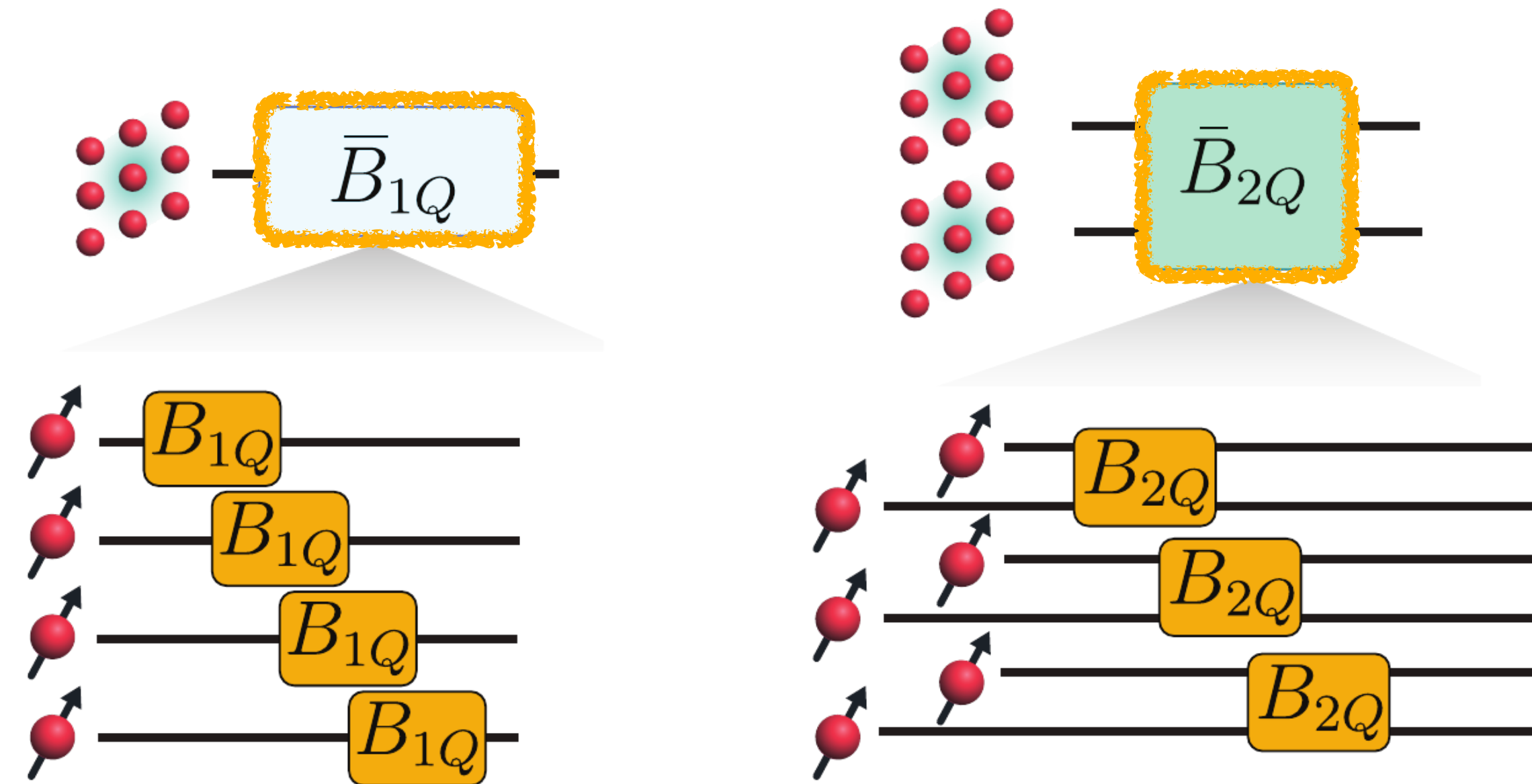
Compile full blind algorithm



Why is this suitable for BQC?

Transversal gate = efficient (few-photon) logical operations

For $[[n, d, k]]$ QEC code



1 logical operation = $\sim n$ photons

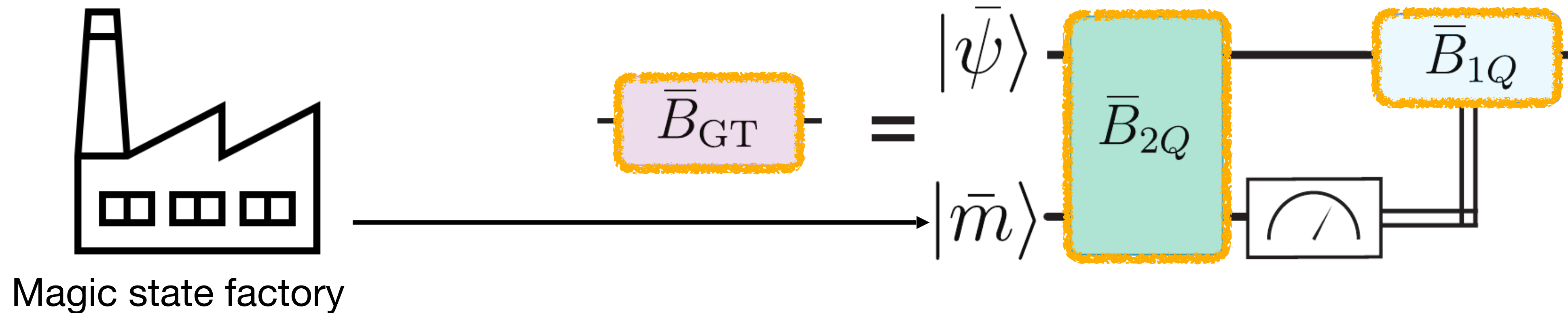
Errors in the delegated gate = local errors in matter qubits

Communication errors are easy to correct and do not spread

Gate Teleportation = efficient (few-photon) magic state teleportation

Transversal gates are not enough
[Eastin-Knill Theorem]

Magic state teleportation gadget, provides the rest



Blind gate teleportation

Why it works:

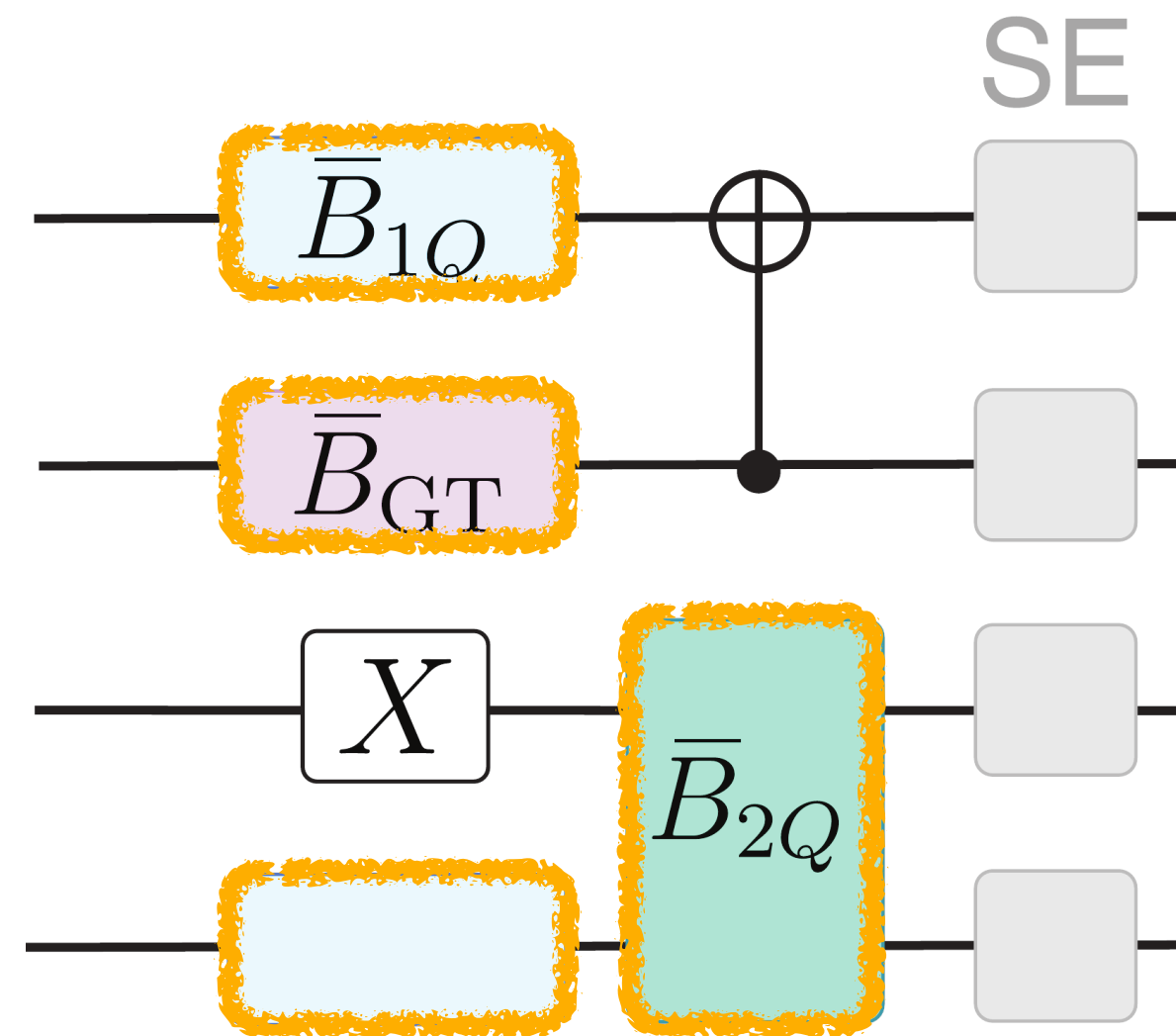
1) Requires same resources as transversal 1Q and 2Q gates

2) **Magic state is generated and distilled without the client's intervention!**

E.g: For the Steane code:

- **all Clifford gates are transversal**
- **T gate injection needs a S gate correction**

Syndrome Extraction (SE) = Efficient and performant EC

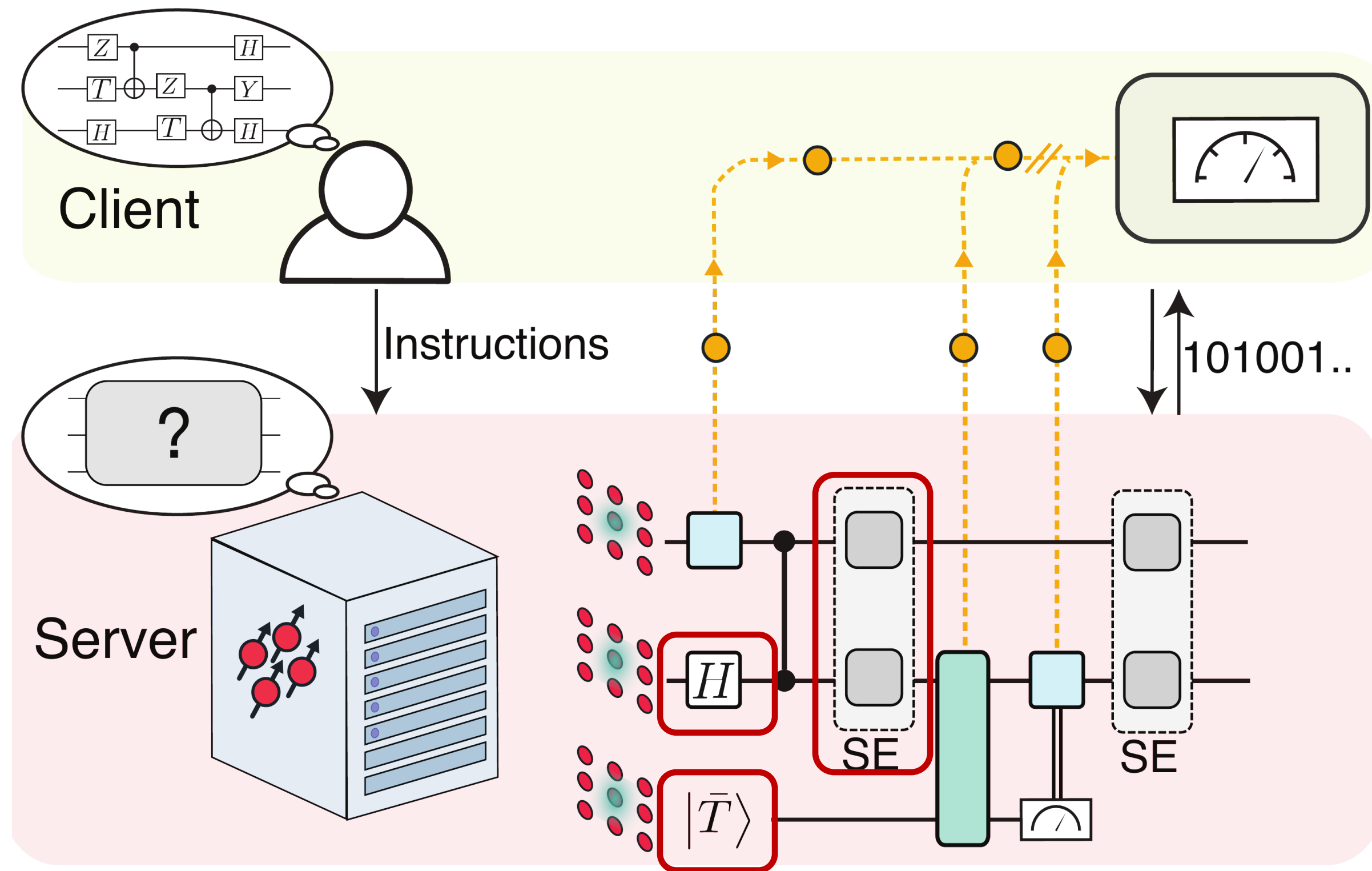


QEC code stabilizer checks can be performed locally on the server!

- 1) Local operations have higher fidelity => error detection is immediately improved!
- 2) Unfortunately, **server cannot do the decoding...**

The Pauli frame alters the outcomes of SE: need client's measurement record to decode
Nevertheless, this is a classical problem that the client can solve

Putting everything together



Photonic qubits saved:

No photons required for stabilizer checks (i.e. SE)

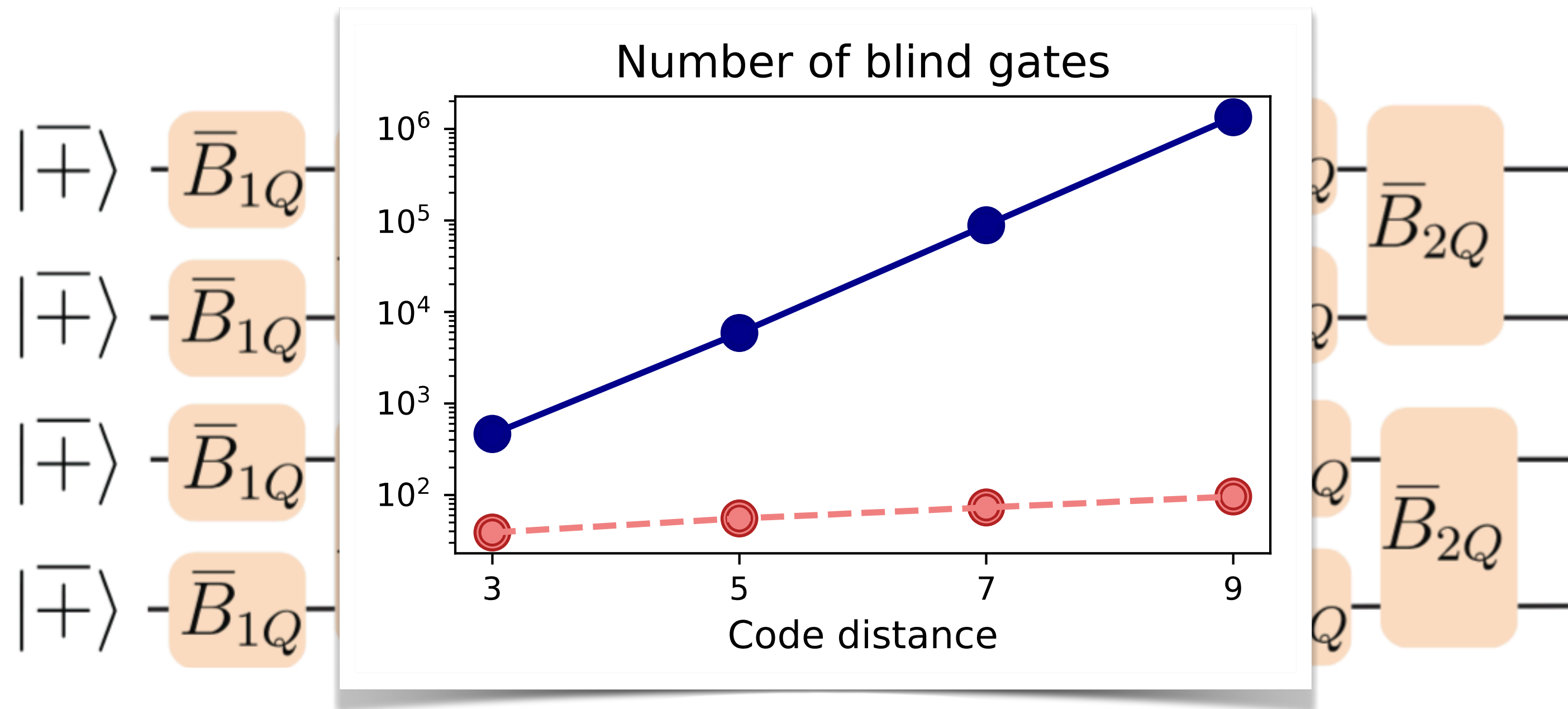
No photons required for non-blind gates

No photons required for magic state generation

Fewer gates, greater tolerance to error

→ **Let's quantify this intuition with a careful analysis of the surface code**

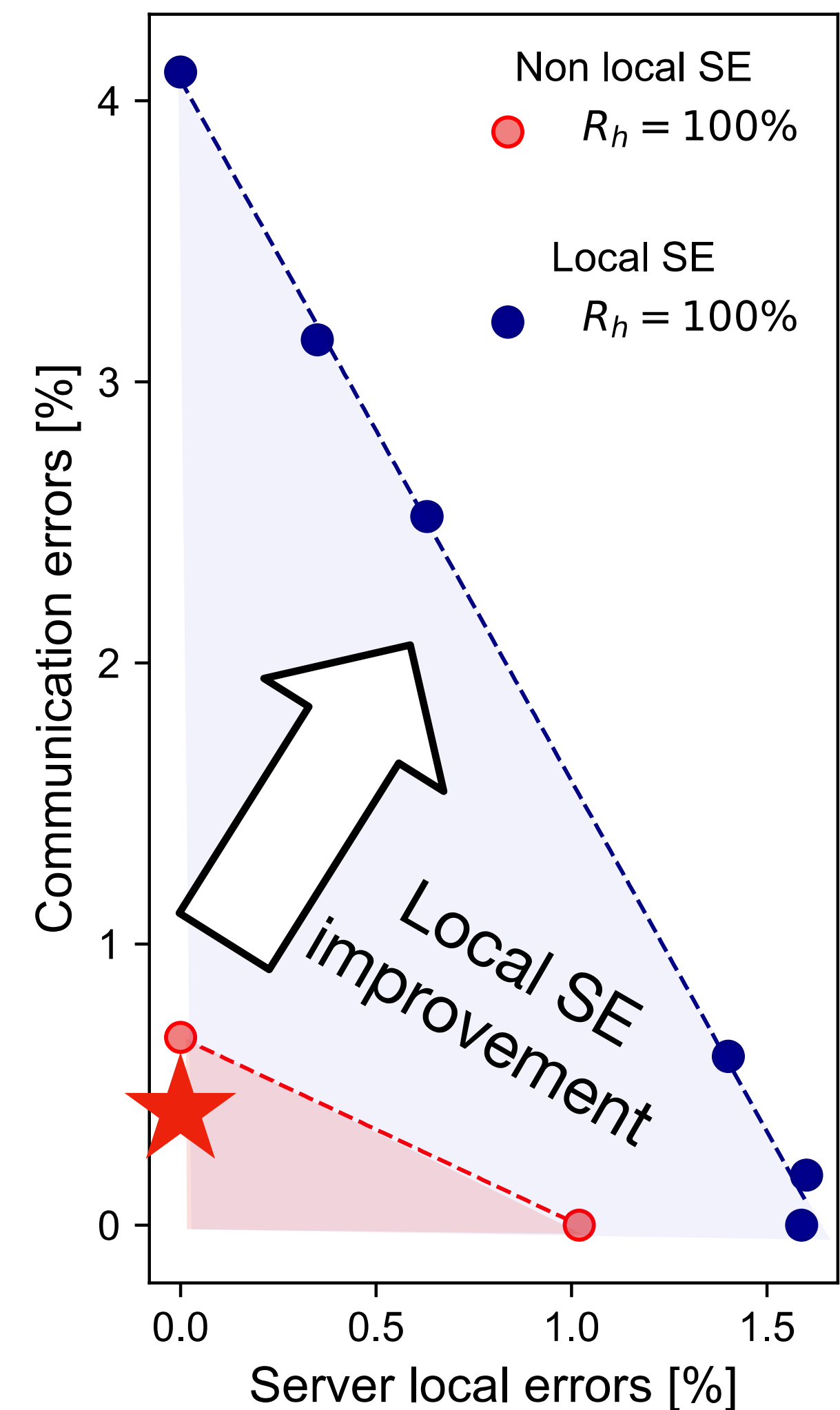
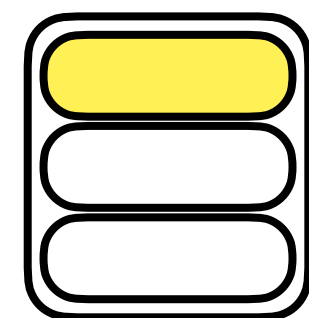
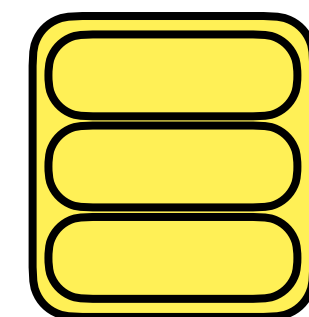
How much do we win?



Compare:

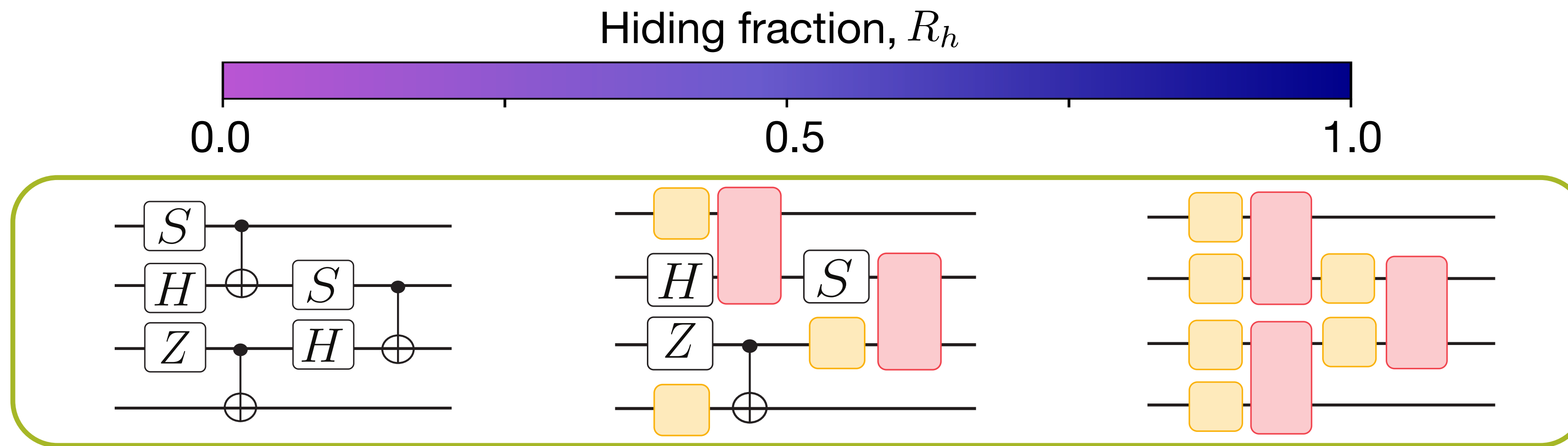
1) Blind Syndrome Extraction (everything ran blindly)

2) Local Syndrome Extraction (SE locally by server)



Changing hiding ratio to improve thresholds

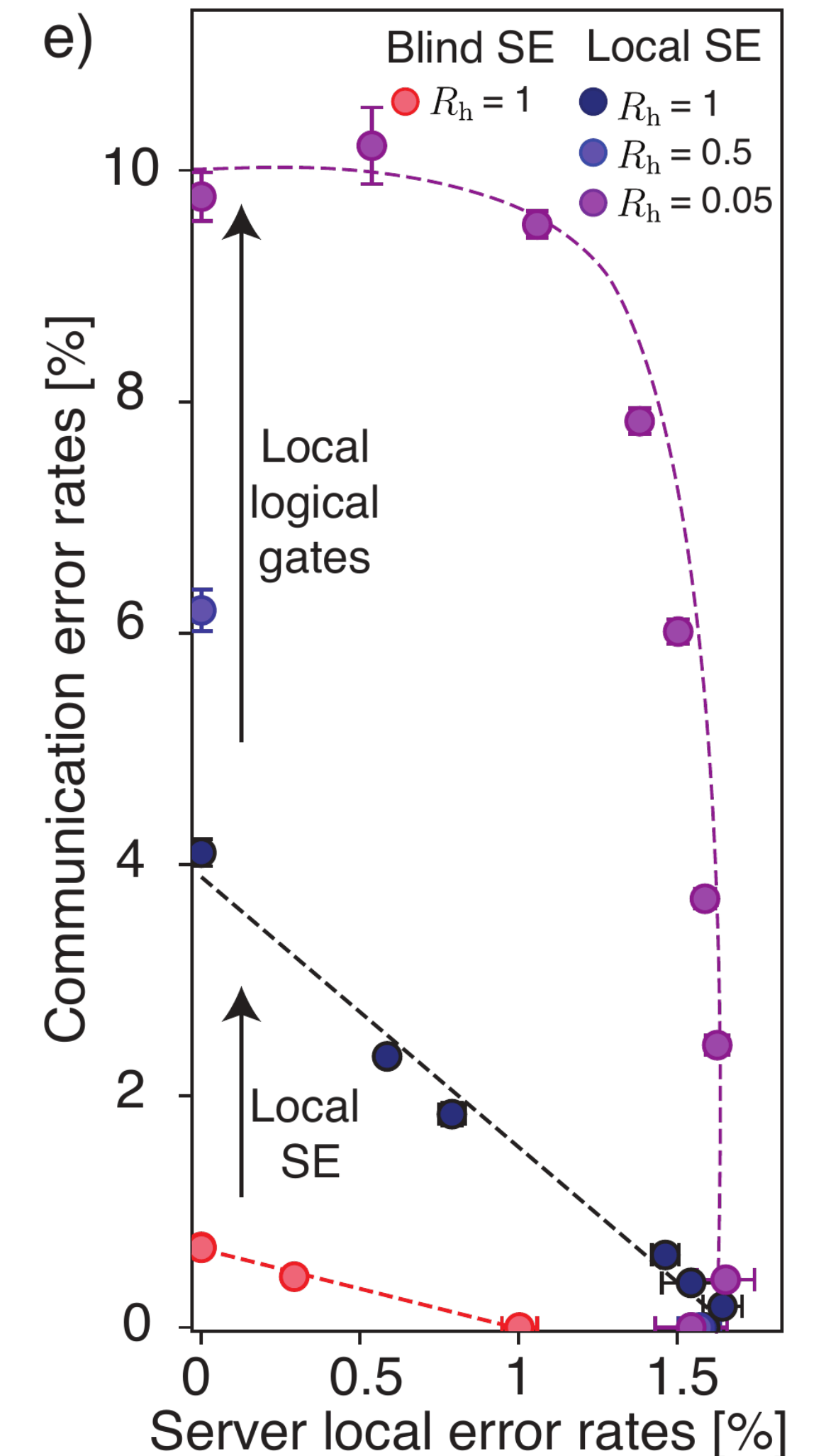
Revealing some of the blind gates, further reduces the number of blind/delegated gates



Further reducing the noisy delegated gates
improves the error correction thresholds

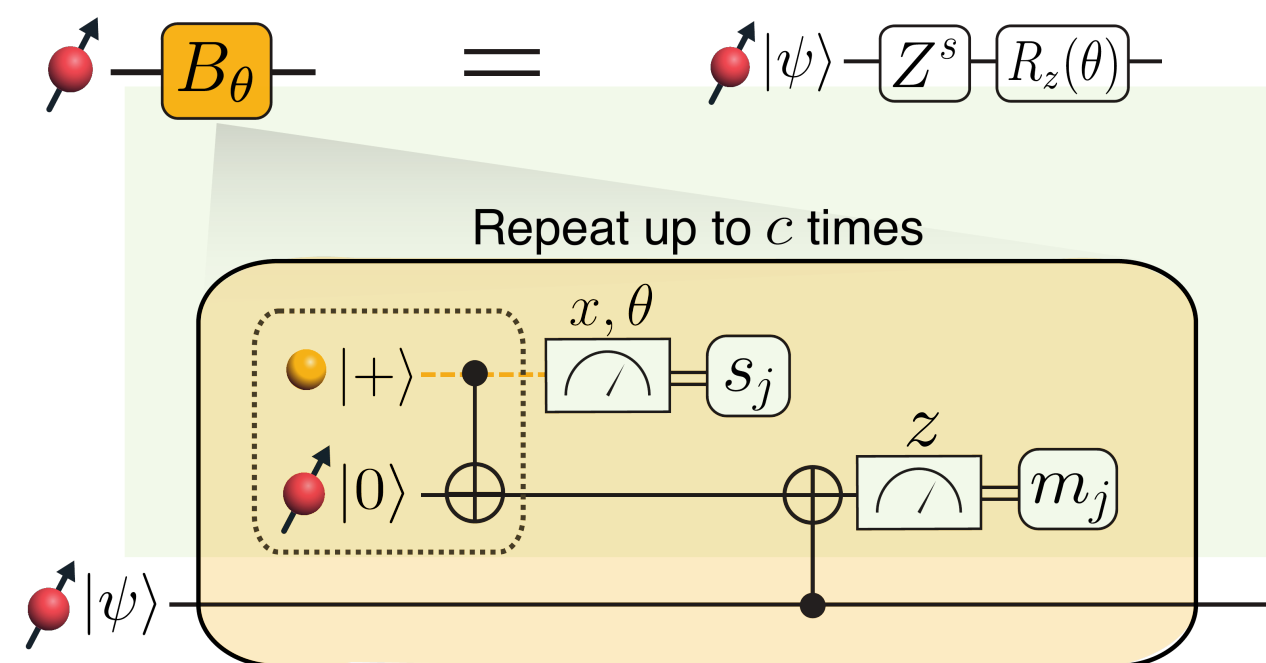
There is a subtle interplay between how efficient error correction is and the circuit compilation

Important open question

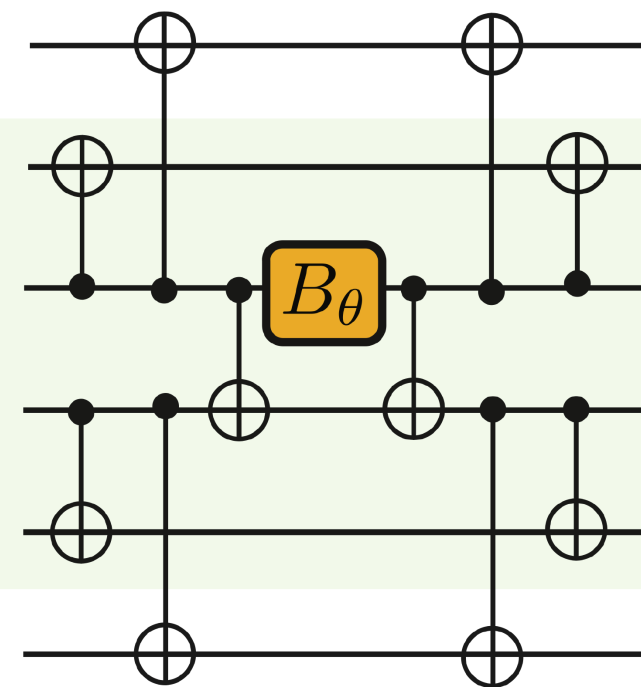


Summary: A better framework for delegated blind computation!

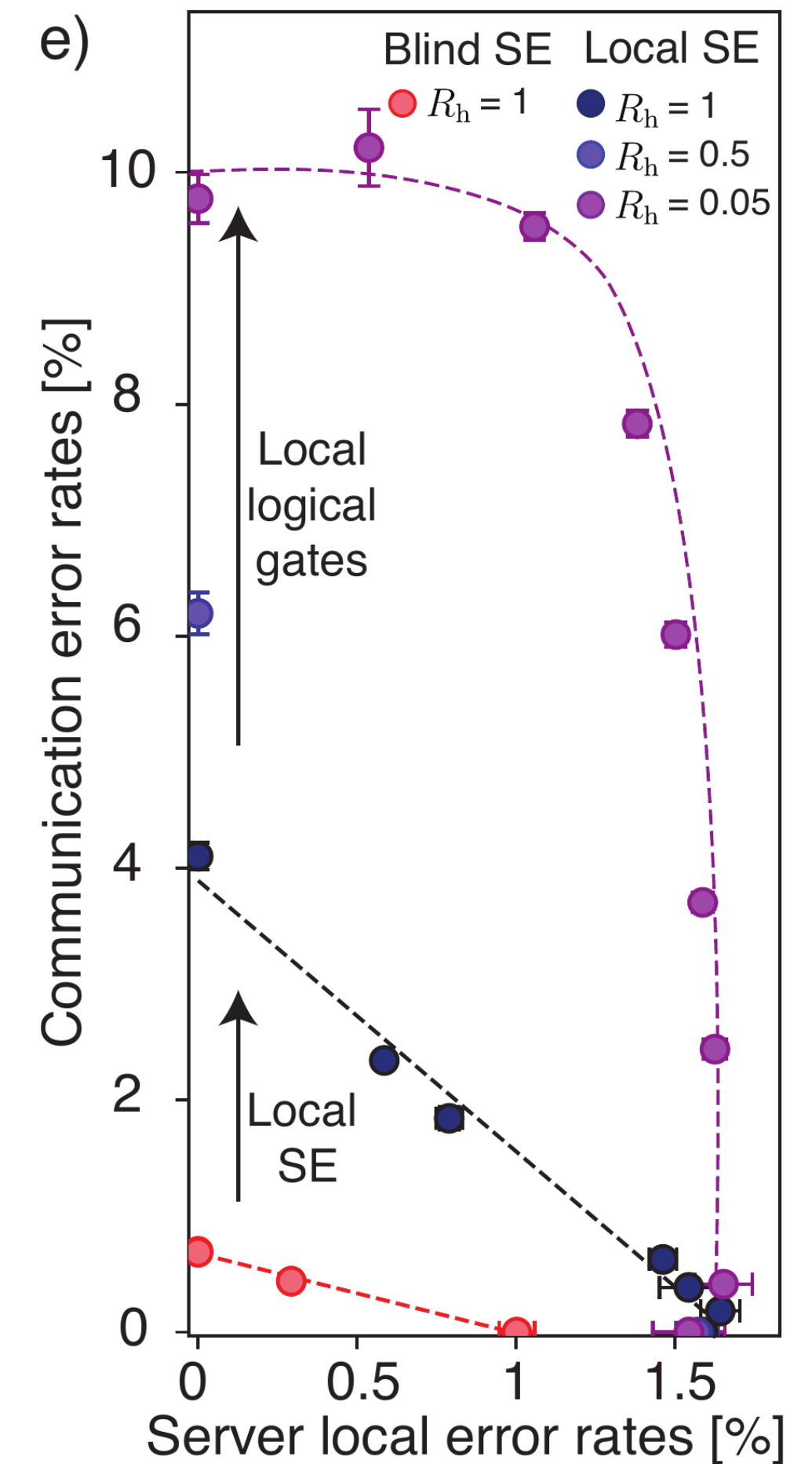
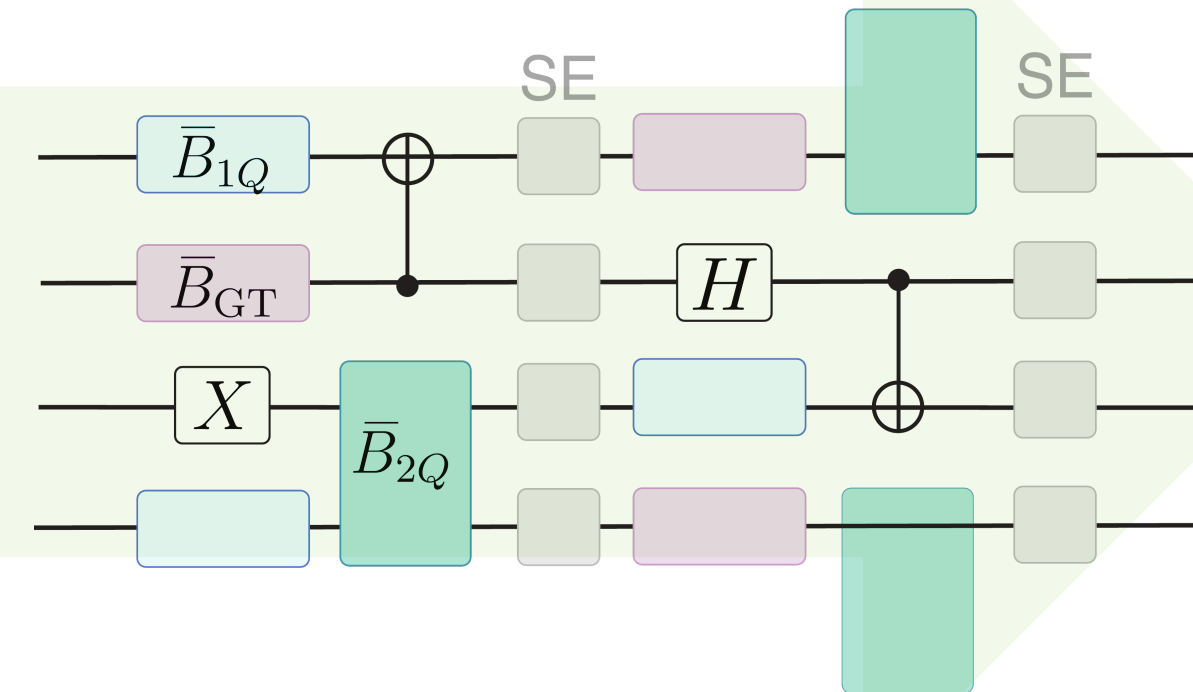
Hybrid Light-Matter Blind QC



Exploring new circuits design possibilities



Fault-tolerant scalable implementation



Outlook: Experiments force theorists to ask better questions!

What does security mean in the context of partially blind circuits?

If the server is malicious, can we confirm they are verify the computation?

What are the best codes for blind computing?

How can we further reduce the communications cost?
(Or why is it impossible?)

Thank you!

Gefen Baranes



Iria Wang



Yan-Cheng Wei



Pieter-Jan Stas



Aziza Suleymanzade



Susanne Yelin



Johannes
Borregaard



Mikhail Lukin

arXiv:2505.21621