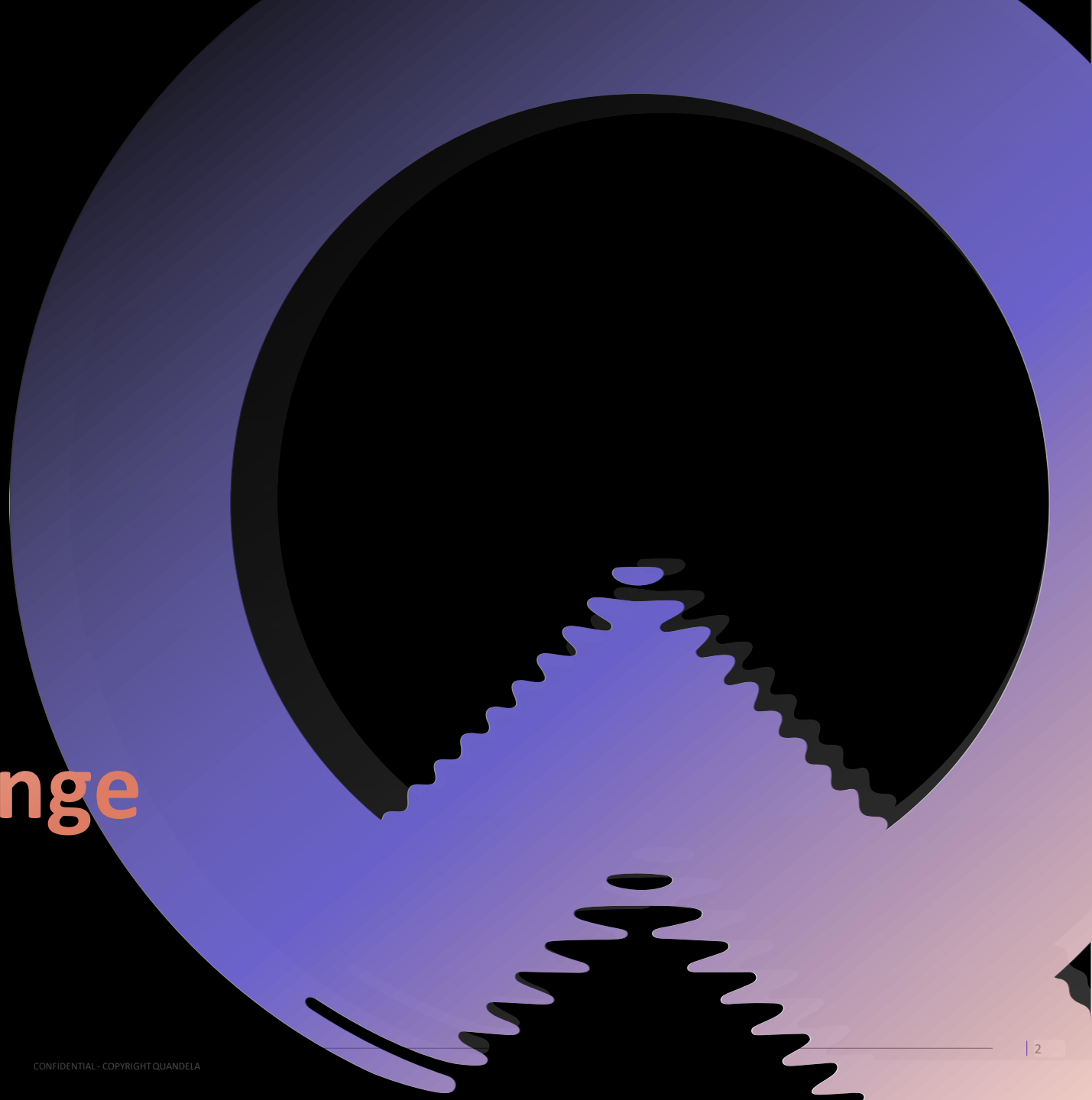
QUANDELA

Scaling Quantum Execution with Automated Circuit Cutting: Applications to Belenos

MATTIA CHIURCO

01.

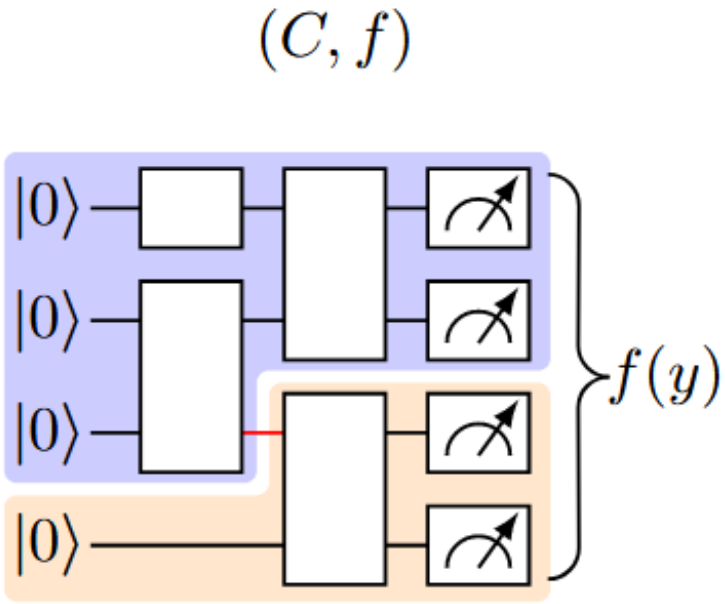
The Scale-Up Challenge



CONFIDENTIAL - COPYRIGHT QUANDELA

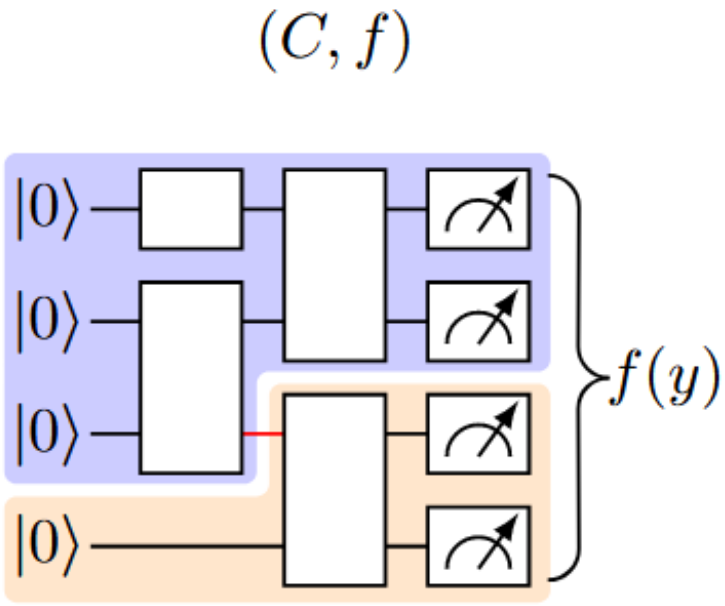
Motivations

- **Hardware limits**: Depth, width, physical space
- **Distributed Computing:** Split computation across multiple smaller devices using quantum communication (state and gate teleportation)
- **Challenge:** Minimize communication via compilation and circuit design



Motivations

- **Problem**: no current machine supports quantum communications, impossible to test algorithms
- **Circuit cutting**: partition a large circuit into smaller subcircuits we can run and recombine classically
- **Cost**: sample complexity blow-up exponentially with the number of cuts
- **Simulate distributed QC:** Good circuit for DQC is also great for circuit cutting.

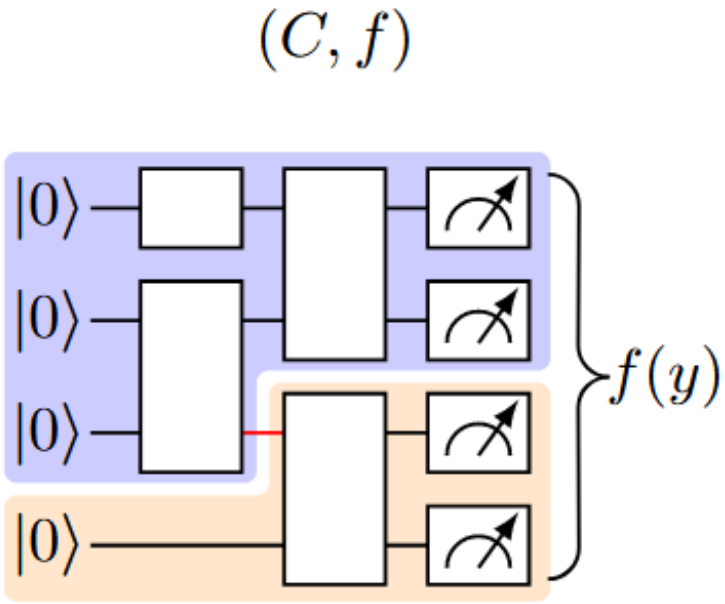


Circuit Cutting: Core Concept

Split a **large** quantum circuit into smaller subcircuits that fit on **limited hardware**, then recombine results classically.

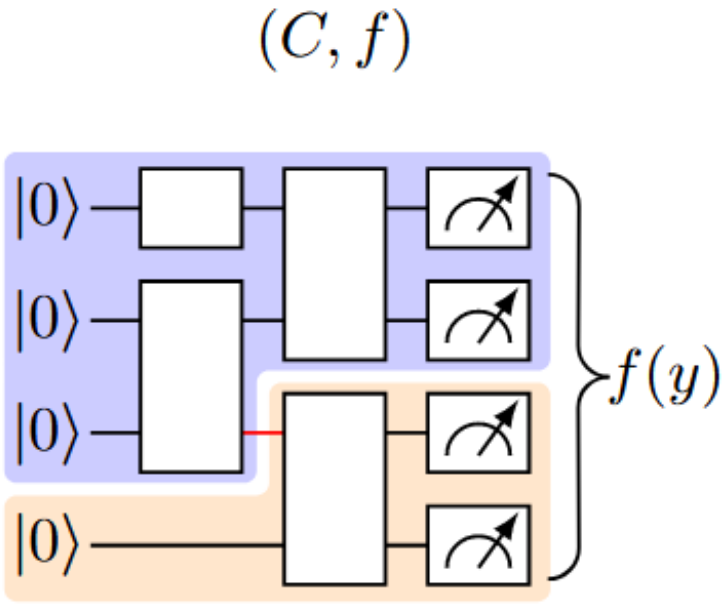
How it works:

- **Partition** the circuit into smaller modules.
- **Cut** the connections (wires or gates) between them.
- **Append** the correct single qubit operations.
- **Recombine** the results



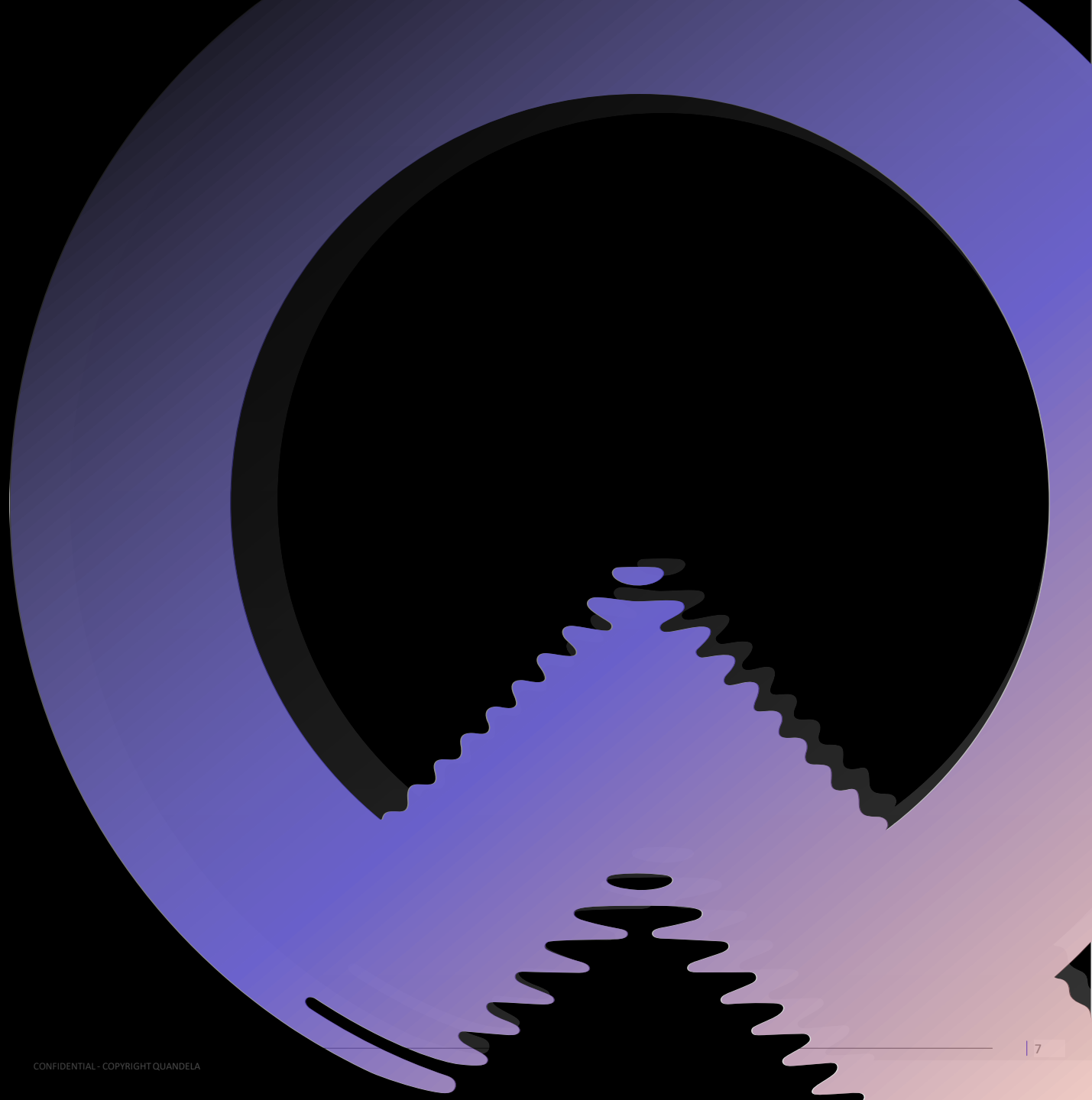
Results

- Implemented circuit cutting on photonic circuits using Perceval for expectation values and sampling
- Designed cut-optimized QUBO ansatz using QLOQ
- Applied circuit-cutting to ansatz to solve optimization problems on the Belenos QPU (MIS, Maxcut, Spin Glass)



02.

Techniques

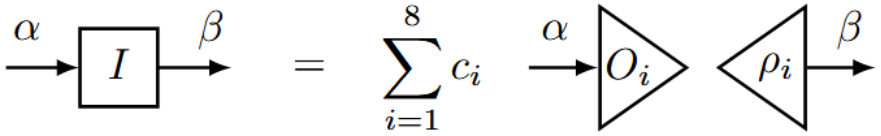


CONFIDENTIAL - COPYRIGHT QUANDELA

Cut Primitives: Wires & Gates

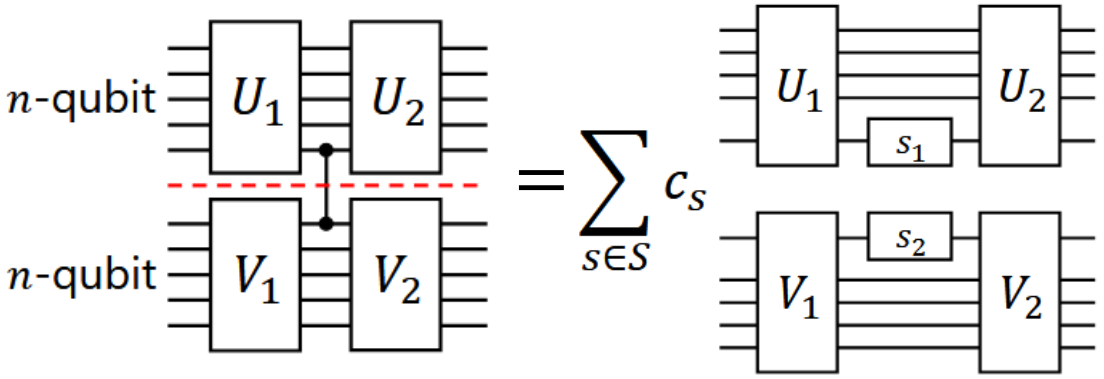
Wire cut

Replace the **identity channel** on a qubit with a weighted sum of “measure & prepare” primitives.



Gate cut

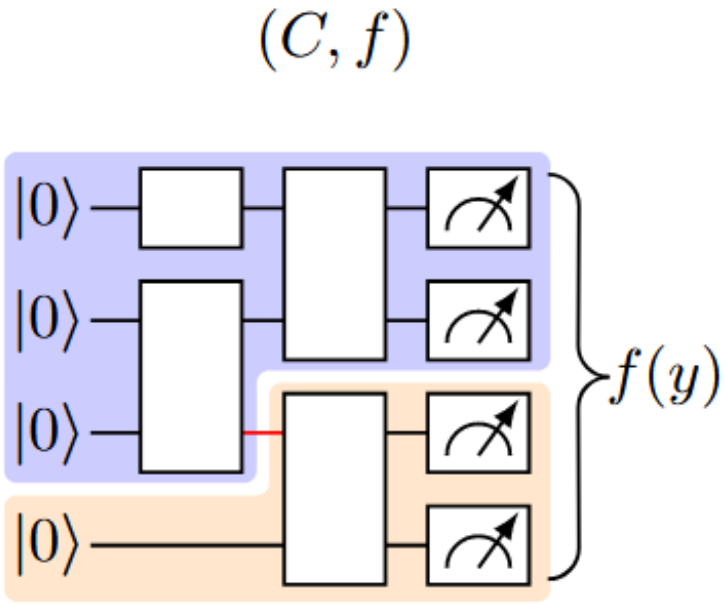
Replace a **two-qubit gate** crossing a partition with a linear combination of local operations.



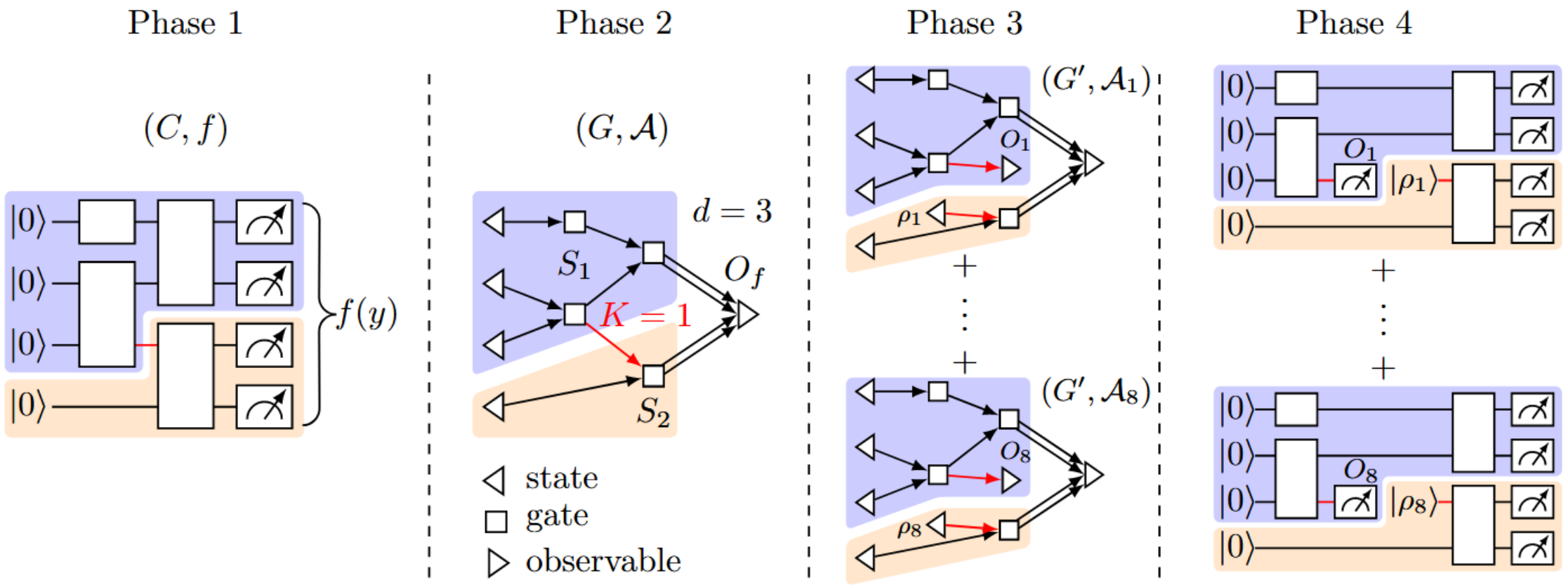
Wire Cutting Protocol (Peng et al.)

Practical recipe:
1. Choose a **basis set** (e.g., {X,Y,Z}).
2. Upstream: **measure** observable(s) producing outcome m.
3. Downstream: **prepare** corresponding state φ_m.
4. Recombine expectation values with signed weights w_m.

Complexity: overhead ∝ (basis terms)^(#cuts)

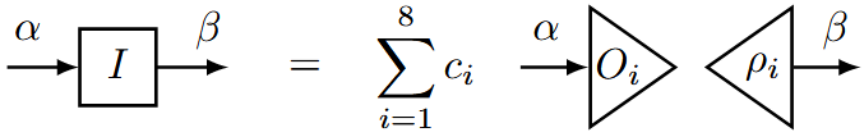


Peng Decomposition



Peng Decomposition

$$
\begin{aligned}
O_1 &= I, & \rho_1 &= |0\rangle\langle 0|, & c_1 &= +1/2,\\
O_2 &= I, & \rho_2 &= |1\rangle\langle 1|, & c_2 &= +1/2,\\
O_3 &= X, & \rho_3 &= |+\rangle\langle +|, & c_3 &= +1/2,\\
O_4 &= X, & \rho_4 &= |-\rangle\langle -|, & c_4 &= -1/2,\\
O_5 &= Y, & \rho_5 &= |+i\rangle\langle +i|, & c_5 &= +1/2,\\
O_6 &= Y, & \rho_6 &= |-i\rangle\langle -i|, & c_6 &= -1/2,\\
O_7 &= Z, & \rho_7 &= |0\rangle\langle 0|, & c_7 &= +1/2,\\
O_8 &= Z, & \rho_8 &= |1\rangle\langle 1|, & c_8 &= -1/2,
\end{aligned}
$$



Optimal Wire Cutting

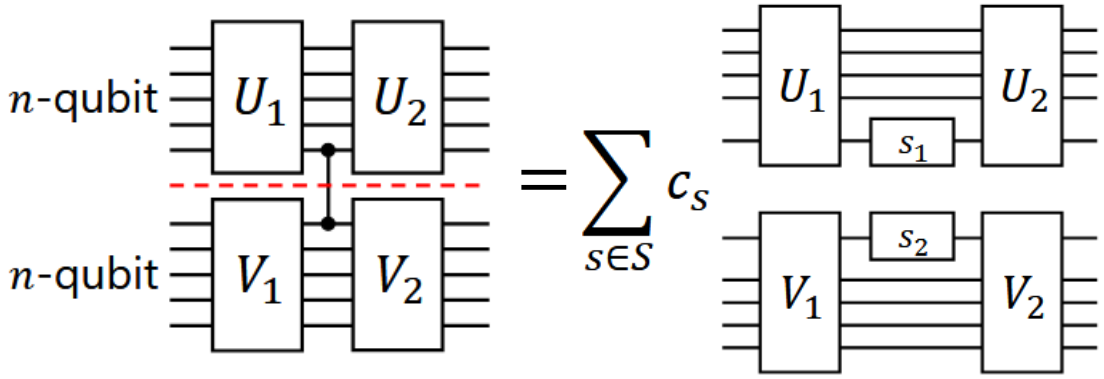
	Without classical comm.		With classical communication	
	Best known	Optimal	Best known	Optimal
(i) Parallel	4^{2n} [13]	4^{2n} [Prop. 4.1]	$\min\{4^{2n}, (2^{n+1}+1)^2\}$ [18]	$(2^{n+1}-1)^2$ [Prop. 4.2]
(ii) Arbitrary	4^{2n} [13]	4^{2n} [Prop. 4.1]	4^{2n} [13]	$(2^{n+1}-1)^2$ [Section 4.3]

Peng et al. vs others:

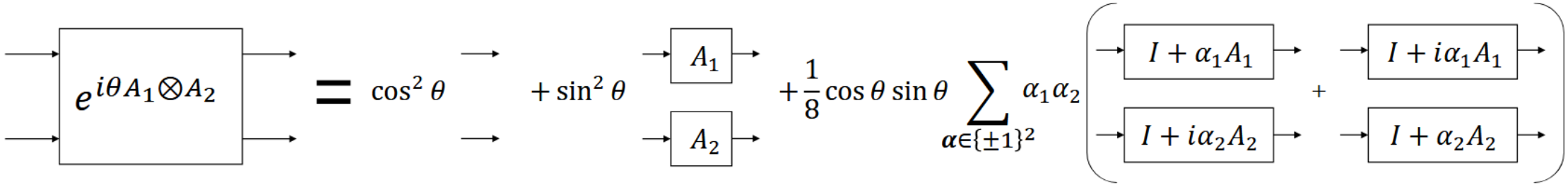
- More costly sample overhead, but no need for multi-qubit gates to implement
- More efficient for photonic QC, where multi-qubit gates are probabilistic

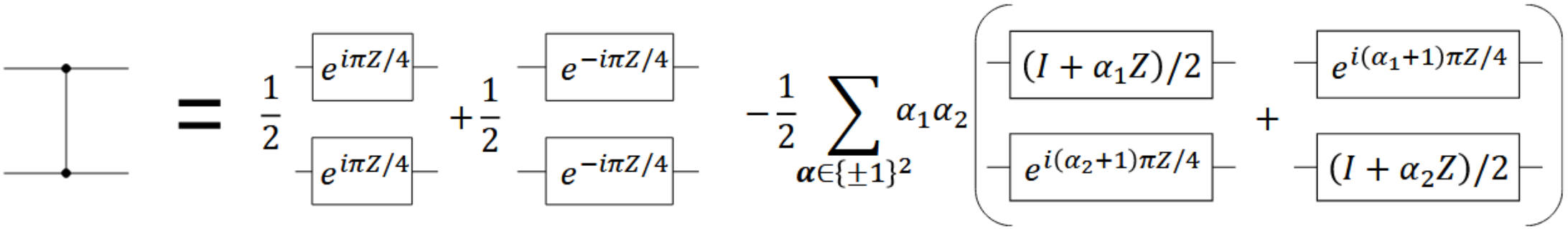
Gate Cutting Mechanism

Goal: Emulate a **2-qubit gate** G crossing a partition $A|B$ using only **local single-qubit** operations and classical reweighting.



Virtual 2-Qubit Gate Decomposition





Gate Cutting Overhead Analysis

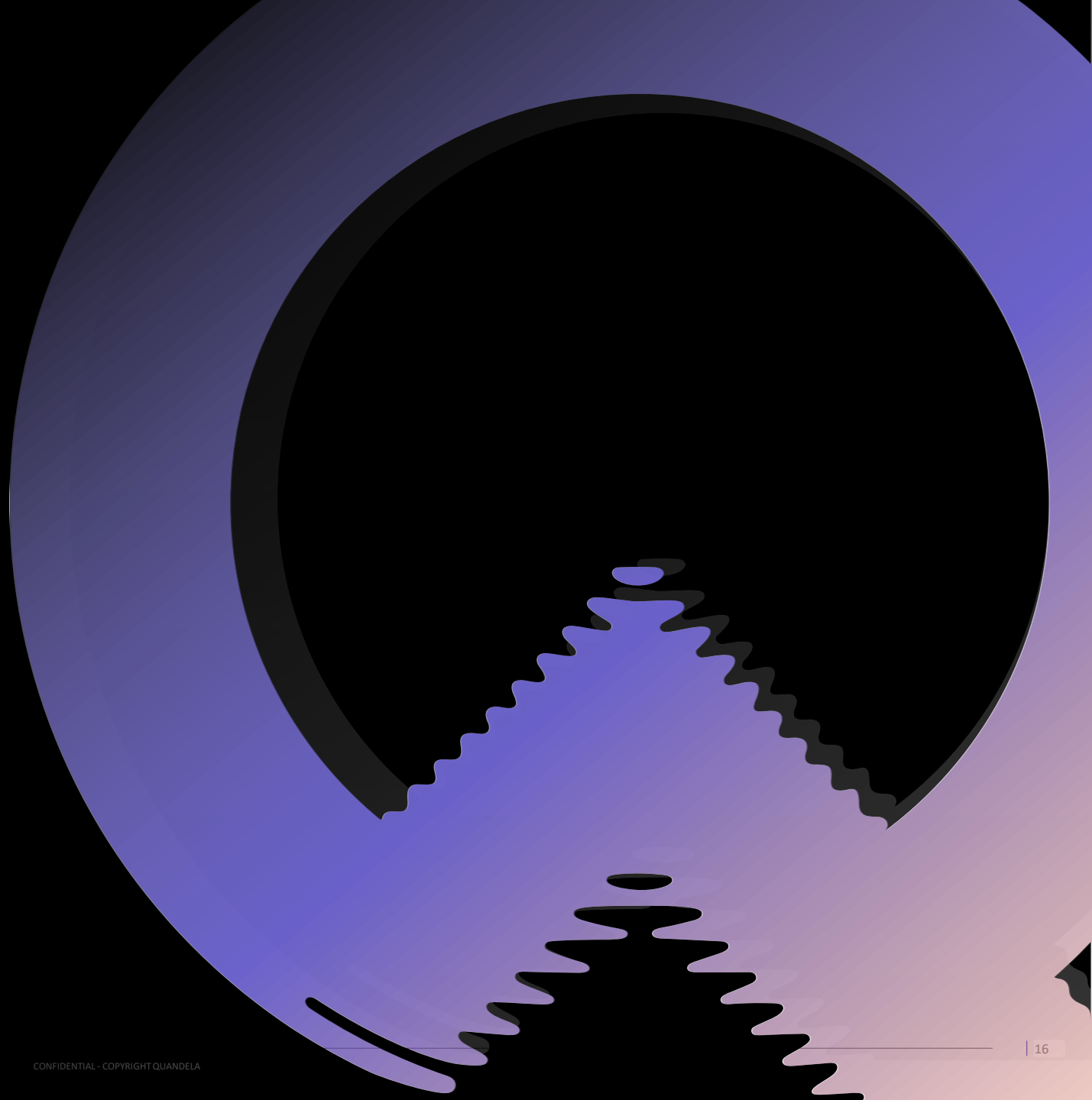
	Wire Cuts		k Gate Cuts		
	Single	k Arbitrary	CNOT	swap	CR(θ)
CC and Ancilla	9	$(2^{k+1}-1)^2$	$(2^{k+1}-1)^2$	16^k	$(\leqslant 4)^k$
No CC, no Ancilla	16	16^k	9^k	49^k	$(1+2\lvert\sin(\theta)\rvert)^{2k}$

Gate cutting vs. heralded gates cost:

- k photonic CNOTs in dual rail: 2k extra modes, 2k extra photons and $(2/27)^k$ success probability (i.e. $(27/2)^k$ sampling overhead on average)
- k CNOTs gate cuts: no extra modes or photons, 9^k sample overhead

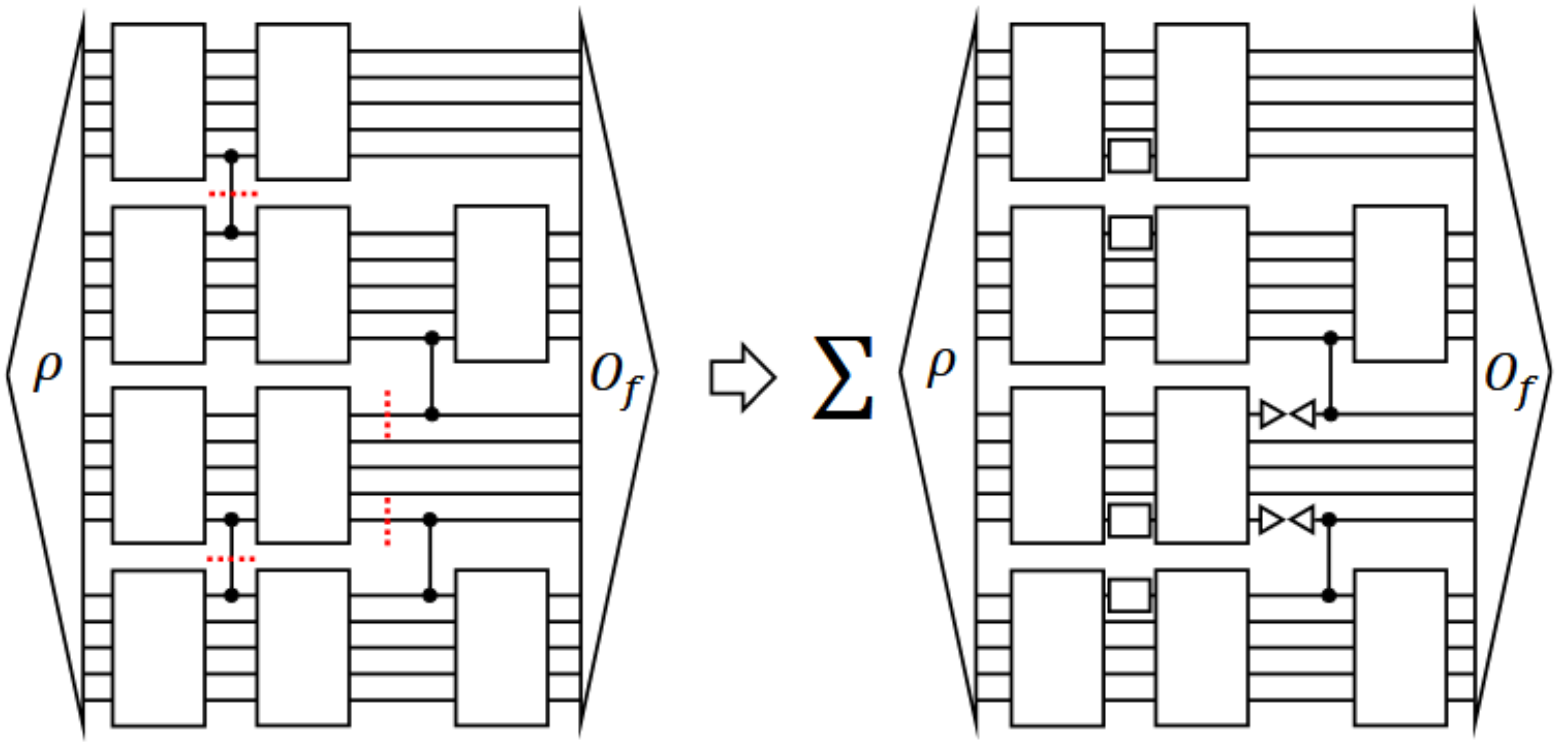
03.

The Automated Cutting Pipeline



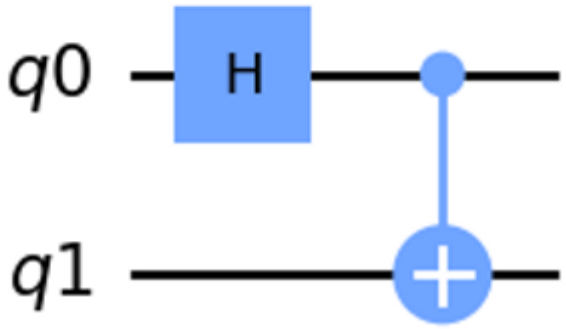
CONFIDENTIAL - COPYRIGHT QUANDELA

Unified Cutting Framework

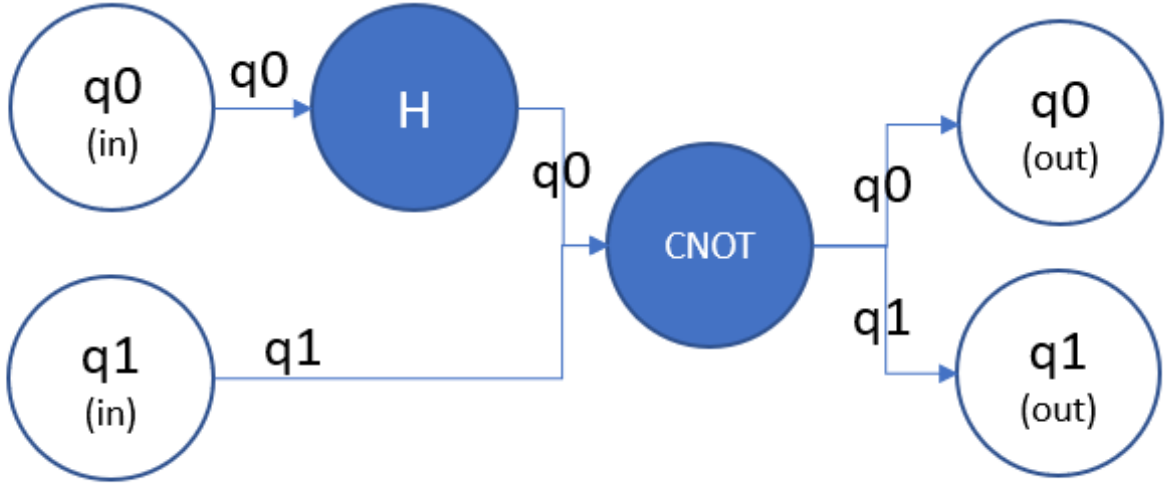


Standard DAG Representation

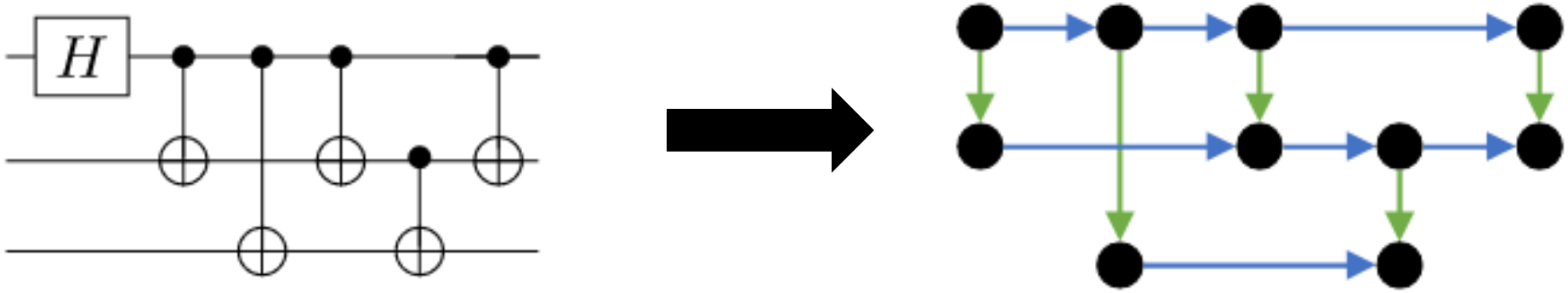
Conventional Circuit Representation



DAG (Directed Acyclic Graph) Representation

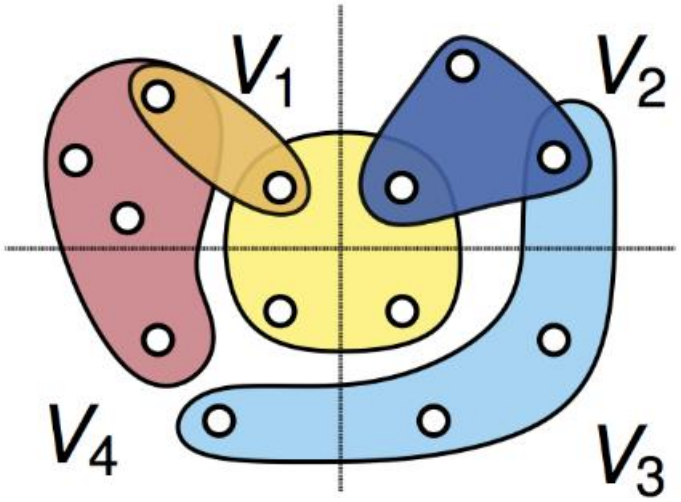


Entanglement Interaction Graph



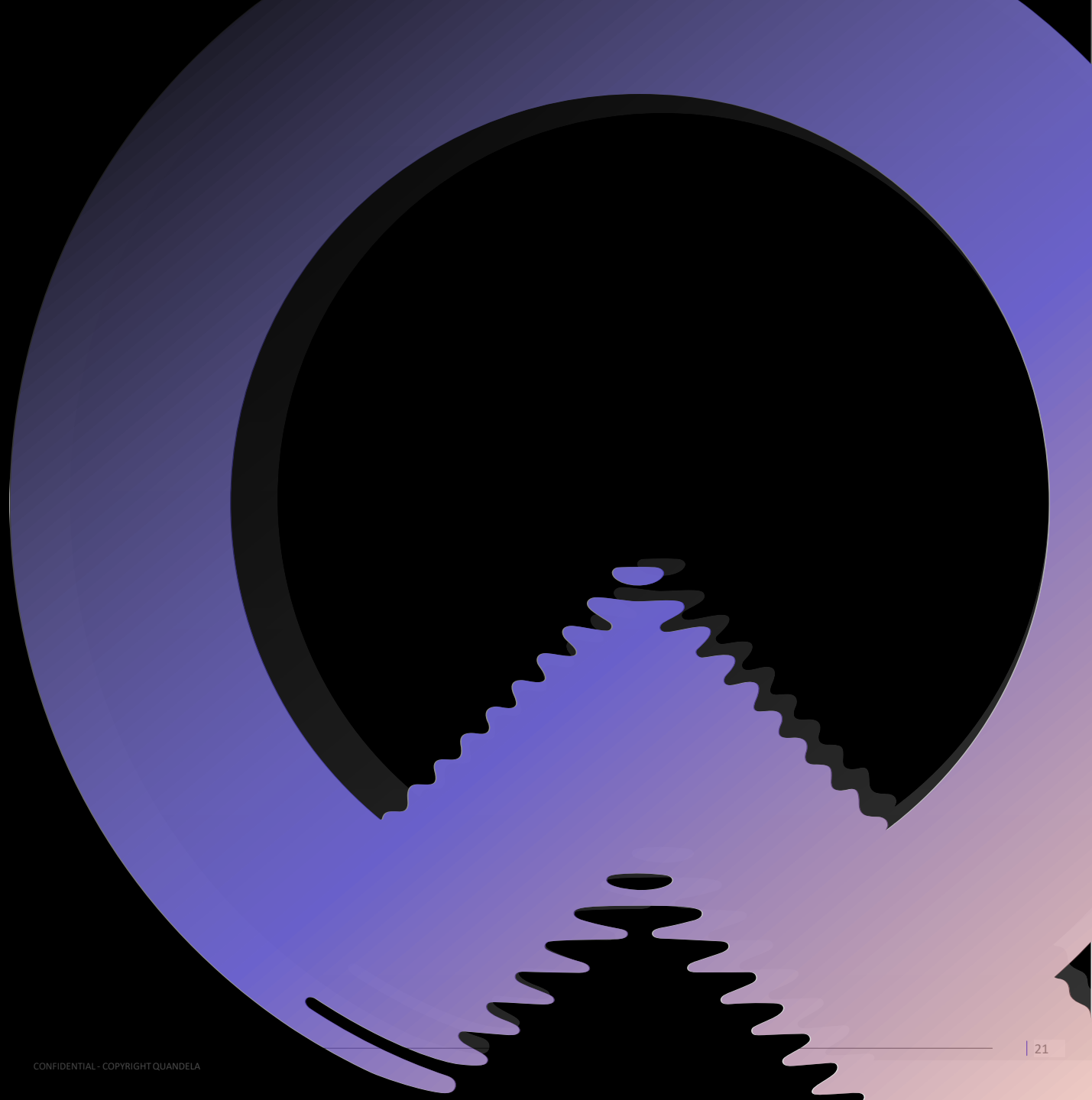
Optimal Partitioning via Hypergraphs

- Build circuit **hypergraph** (ops = vertices, dependencies = hyperedges)
- **Run Mt-KaHyPar** to split into k balanced fragments



04.

Applications



CONFIDENTIAL - COPYRIGHT QUANDELA

Optimization Benchmark Suite

MaxCut

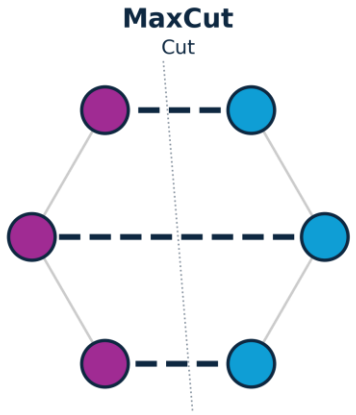
- Split nodes into two sets to maximize edges between them
- Global optimum emerges from conflicting local edge choices

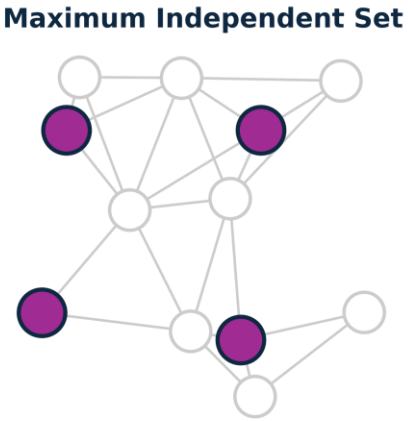
MIS (Maximum Independent Set)

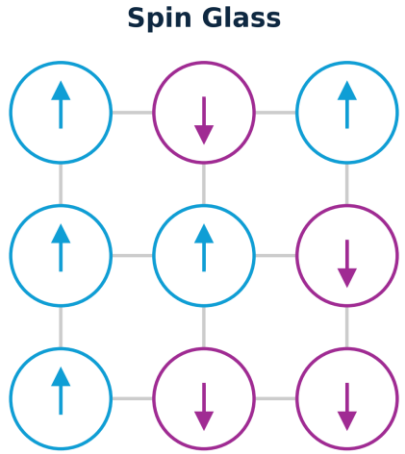
- Choose the largest set of mutually non-adjacent nodes
- Selecting a node forbids all its neighbors

Spin Glasses

- Spin with random $+1/-1$ interactions on a graph
- Frustration creates many competing low-energy configurations





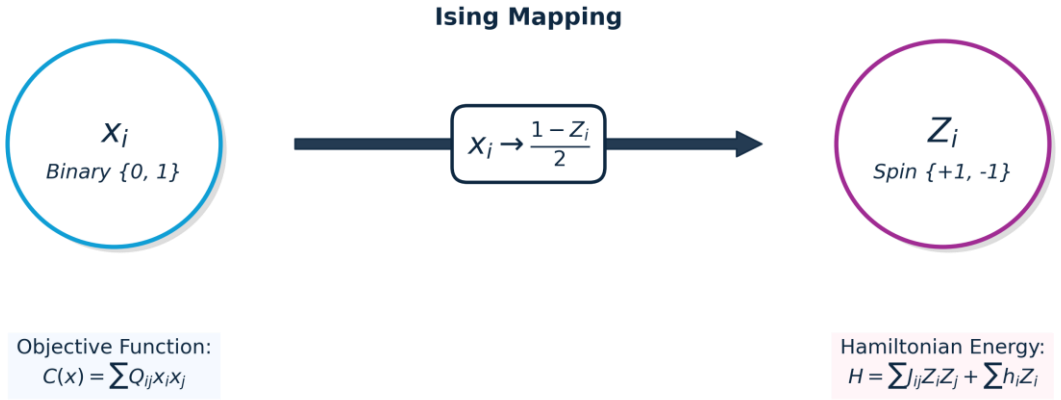


Problem Encoding: Ising Formulation

- **QUBO:** Quadratic Unconstrained Binary Optimization
- **Encoding:** All previous graph problems map to a Ising Hamiltonian

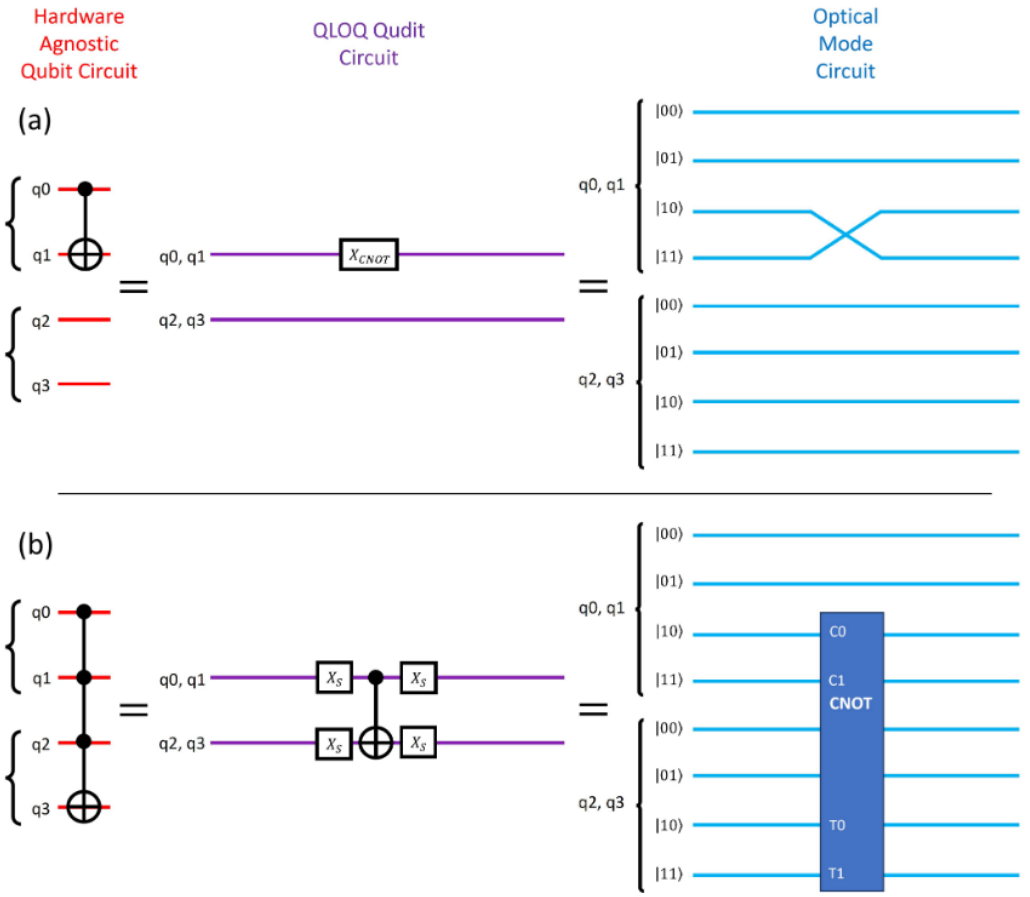
$$H = \sum_{i,j} J_{ij} Z_i Z_j + \sum_i h_i Z_i$$

- **Goal:** Find the bitstring z that minimizes the energy $\langle H \rangle$

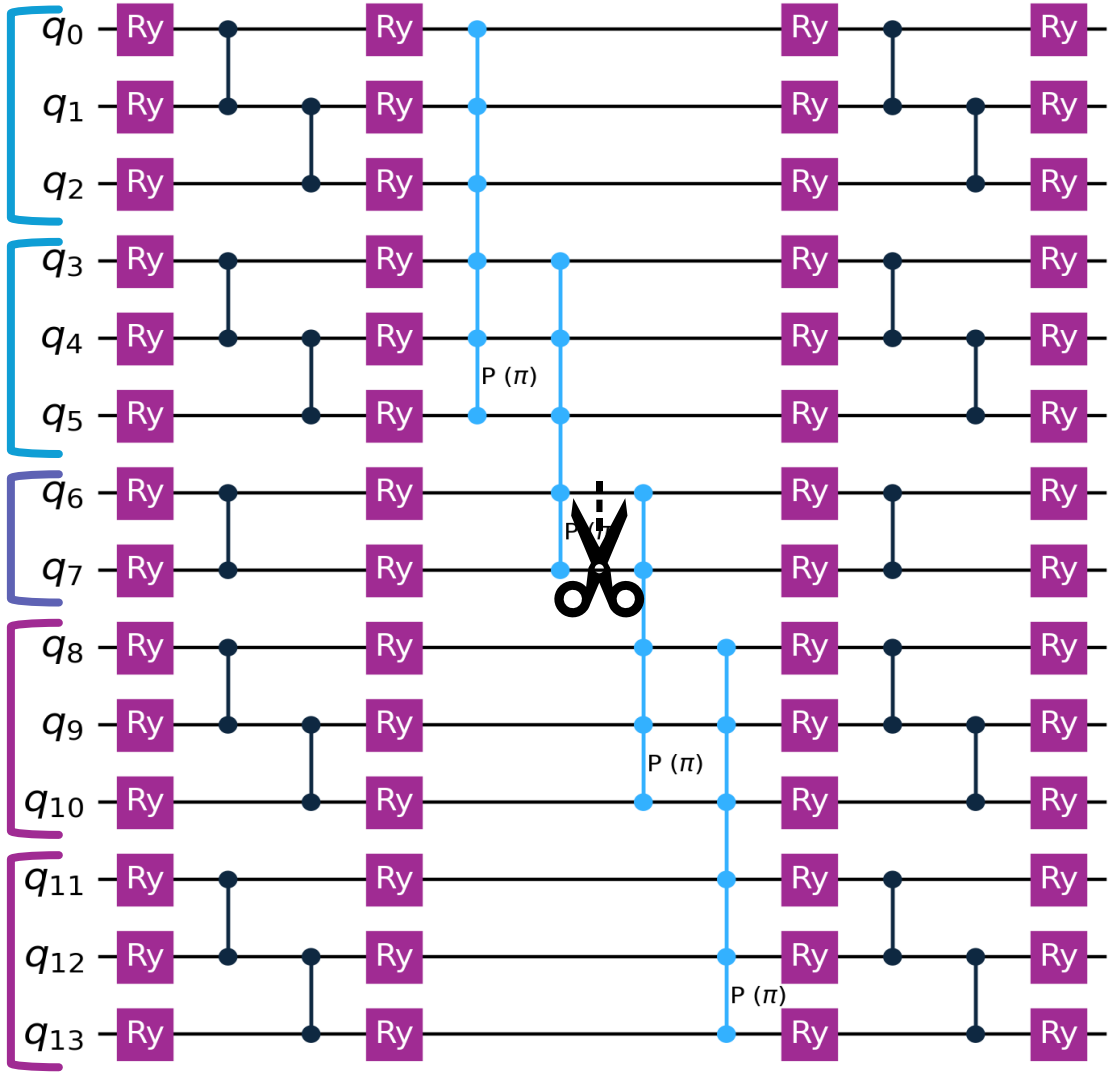


The QLOQ Encoding Paradigm

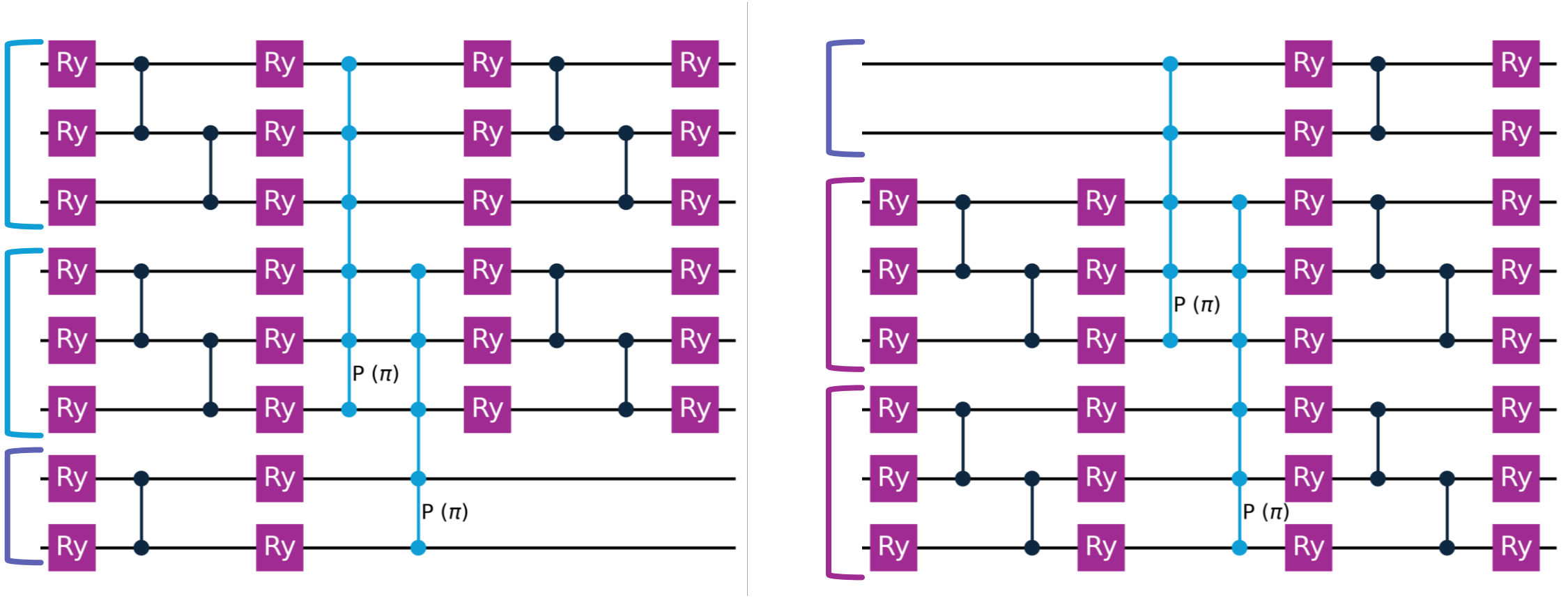
- **Generalization of dual-rail**: 1 photon in 2^n modes encodes n qubits
- Intra-group 2-qubit gates become **deterministic** mode permutations/rotations
- Only inter-group entanglement is probabilistic.



The Belenos Ansatz Architecture



Partitioned Sub-circuits Layout

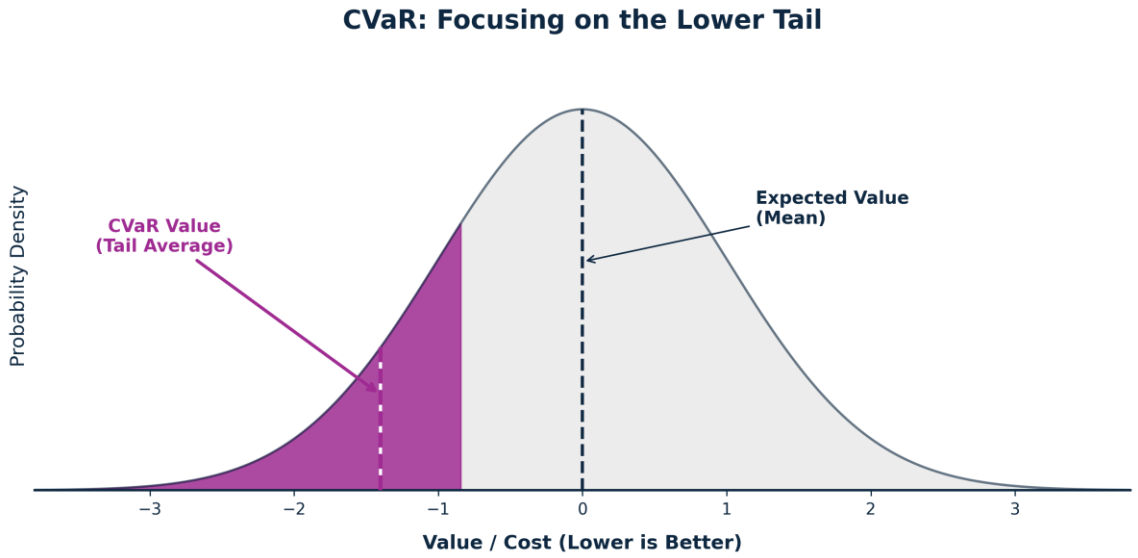


Enhancing Readout with CVaR

The Concept: Optimize the average of the lower tail instead of the full mean

Why use it?

- Accelerates convergence to the ground state in variational algorithms (e.g. VQE)
- **Crucial for cutting**: Acts as a filter. CvaR makes the reconstruction more robust.



Cutting Beyond Expectation Values: Sampling

Why sampling?

- CVaR requires individual bitstrings, not just expectation values.

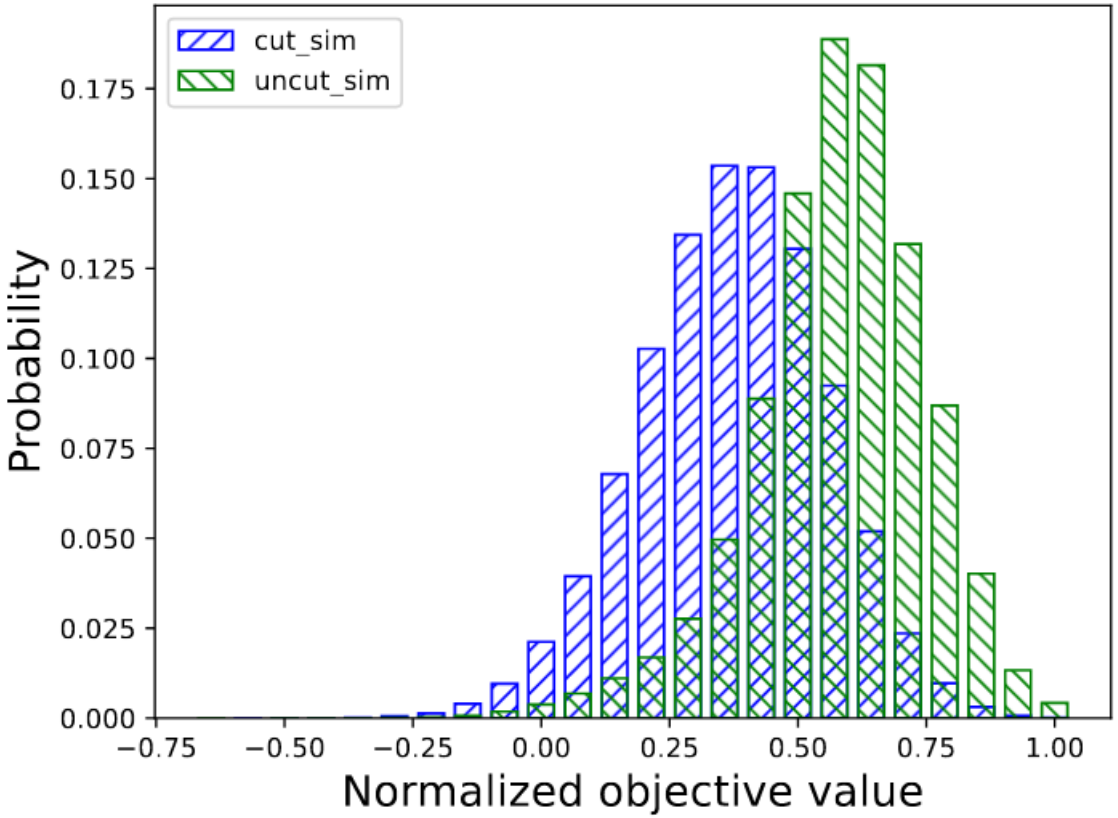
The challenge:

- Circuit cutting cannot directly be used for sampling.

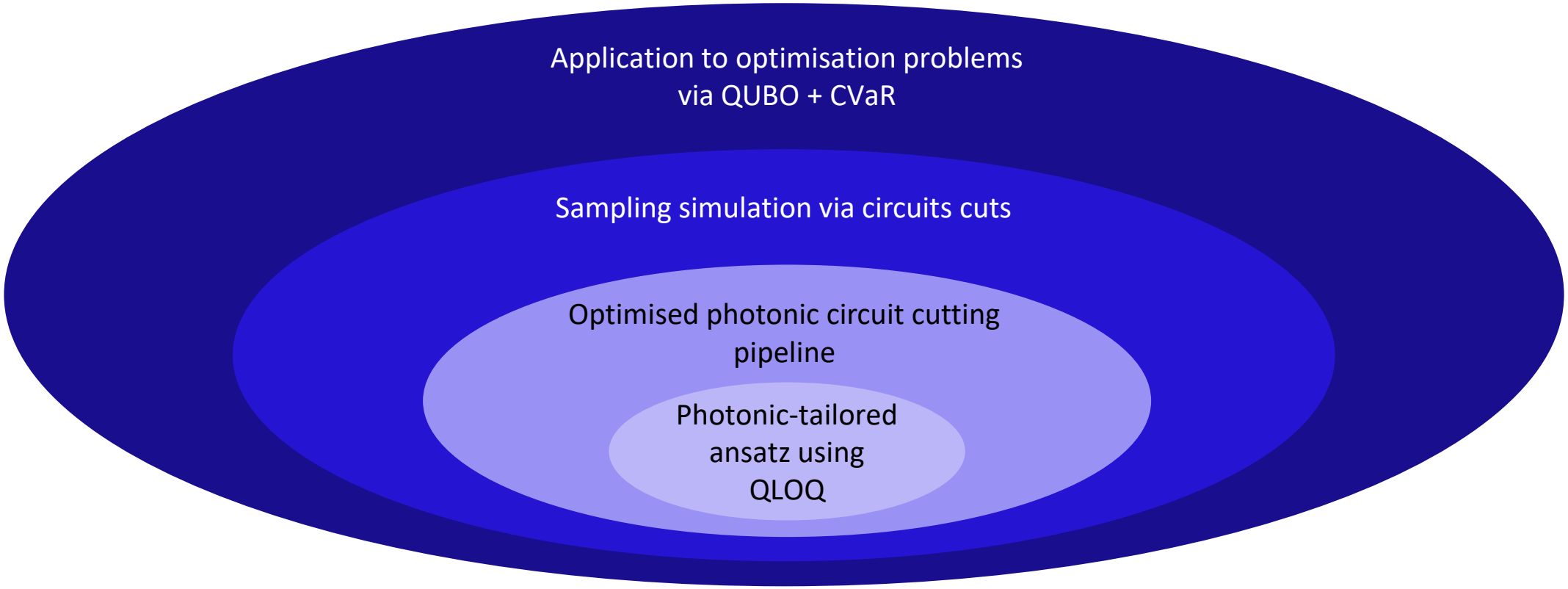
Our approach:

- Sample from a modified distribution p̃(s).
- **Guarantee:** All possible outcomes remain accessible (support preserved).

$$\tilde{p}(s) \geq \frac{1}{\kappa}\, p(s)$$

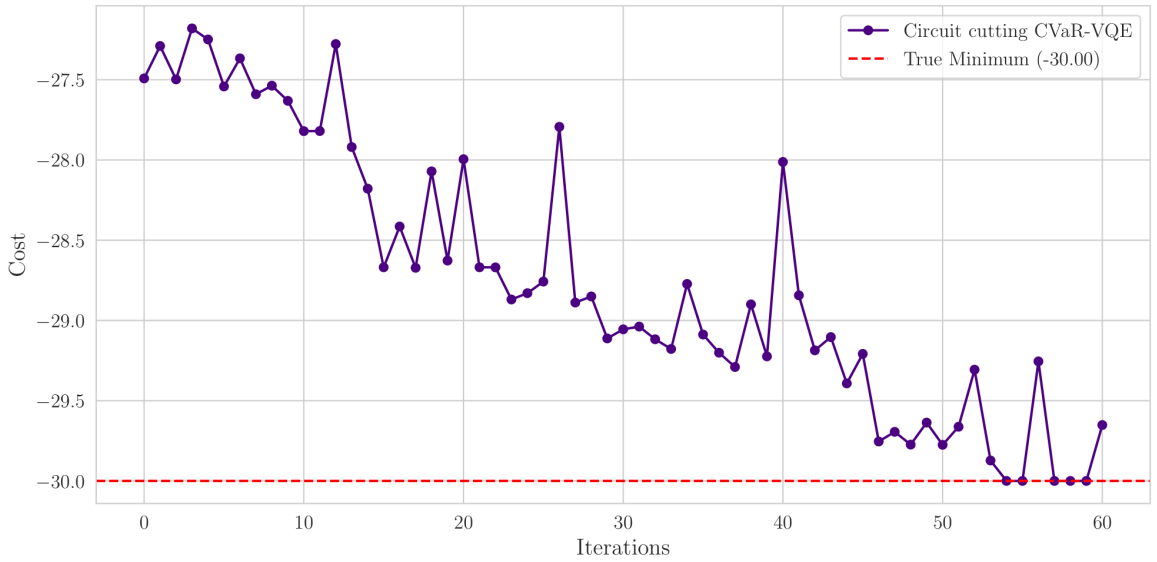


Overview of Techniques

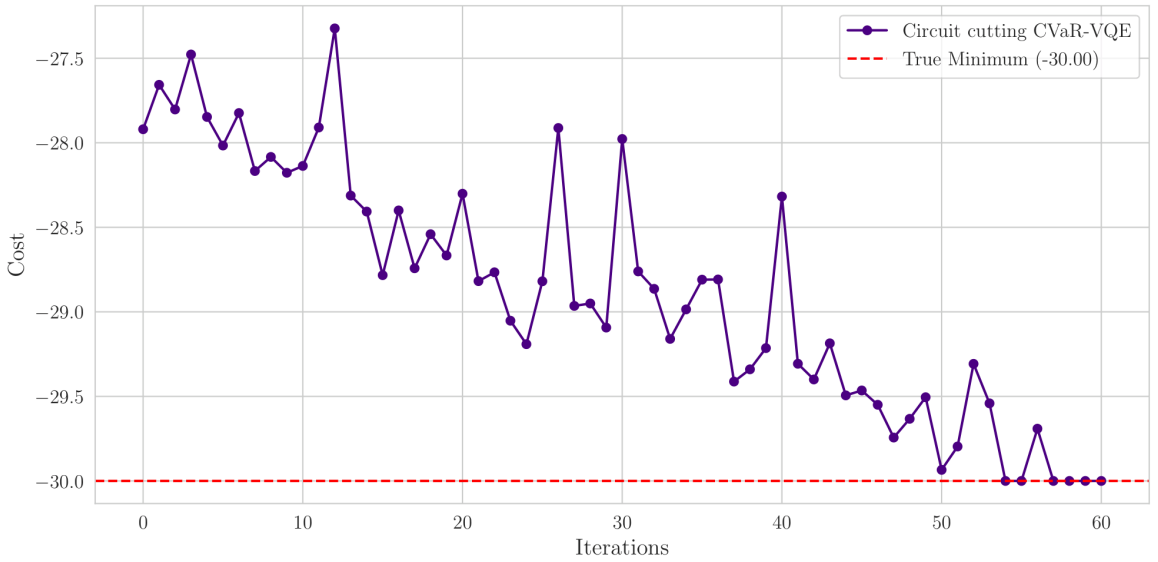


MaxCut Optimization: Simulation vs. Belenos Hardware

State vector simulation

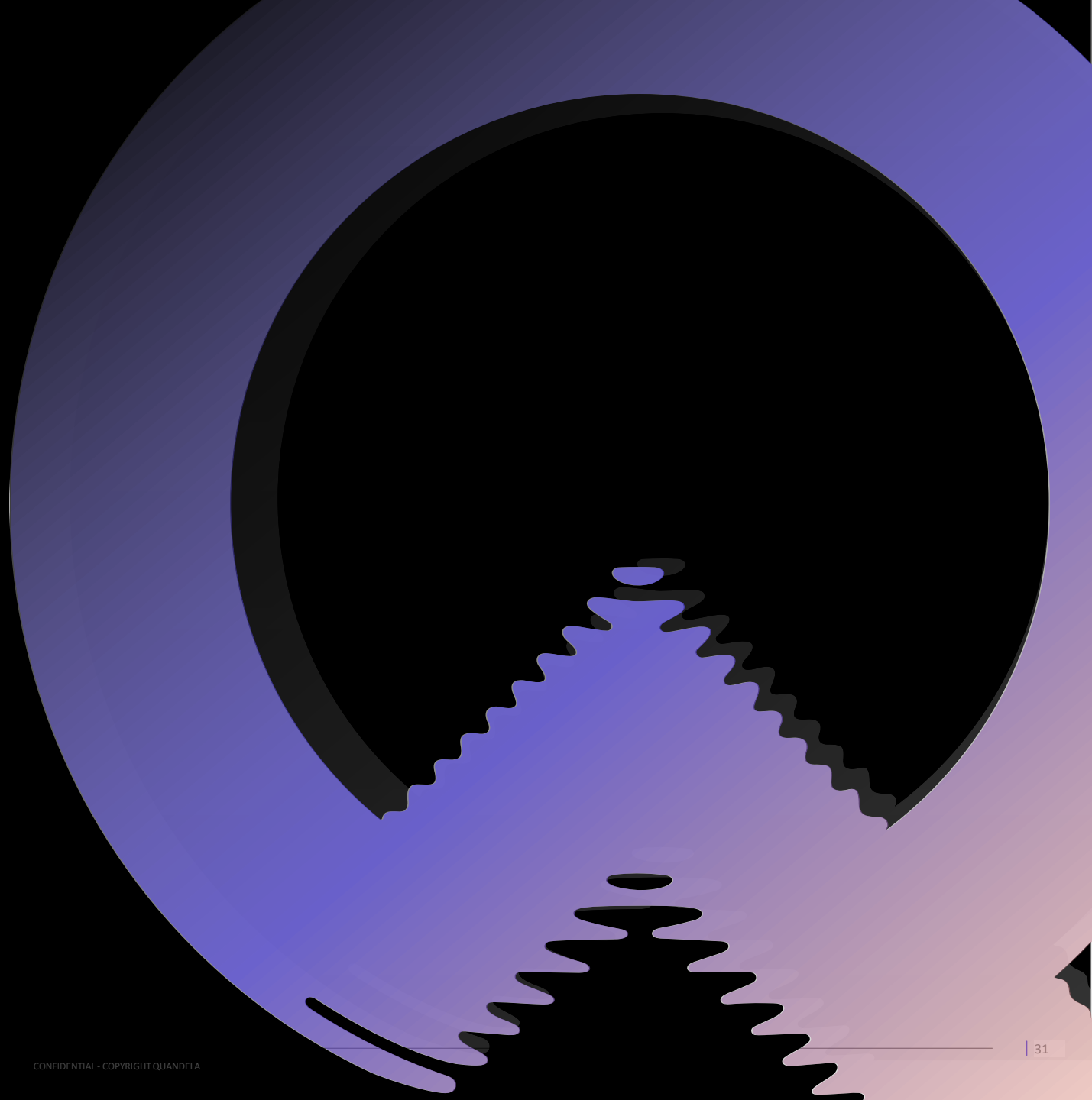


Belenos simulation



05.

Conclusion



CONFIDENTIAL - COPYRIGHT QUANDELA

Conclusion & Outlook

- **Automated Pipeline:** Built a complete tool for wire and gate cutting, integrated with Perceval, that automatically finds good ways to split quantum circuits.
- **Optimization Pipeline:** Combined circuit cutting, QLOQ, and CVaR to solve larger optimization problems using smaller quantum circuits.
- **Results on Belenos:** Successfully tested our approach on the Belenos quantum processor, running optimization problems with more qubits than the hardware can directly support.
- **Path Forward:** Add realistic noise from quantum communication to study how our approach performs in distributed quantum computing.



CONFIDENTIAL - COPYRIGHT QUANDELA