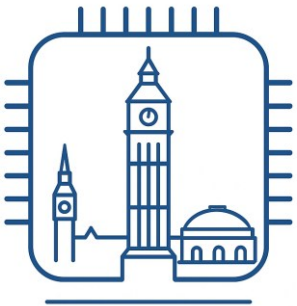


Efficient Circuit Management and Scheduling in Multi-Node Quantum Systems with Dynamic Links

Zefan Du, Wenrui Zhang, Jake Gesseck, Wenqi Wei, Juntao Chen, Tao Han, Zhiding Liang, Ying Mao
ASAP 2026



FORDHAM UNIVERSITY
THE JESUIT UNIVERSITY OF NEW YORK



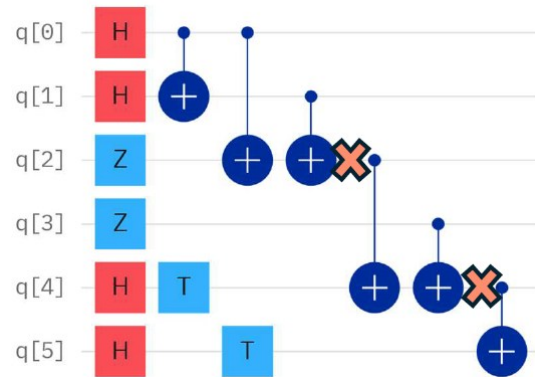
PART 1

Background & Motivation

How to scale quantum circuits

Scaling circuits creates a two-sided trade-off

Circuit cutting



Classical reconstruction

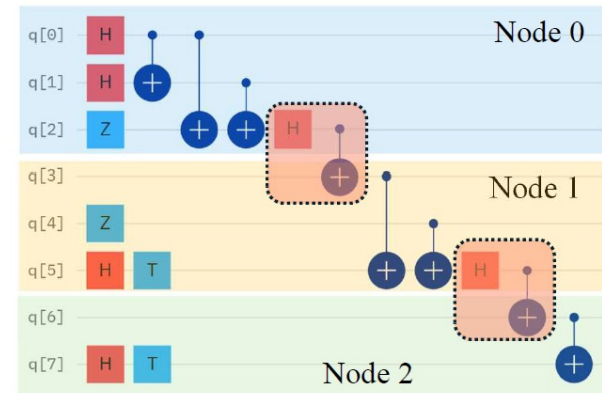
Cut count

K

Cost

$O(4^K)$

EPR-linked system



Quantum communication

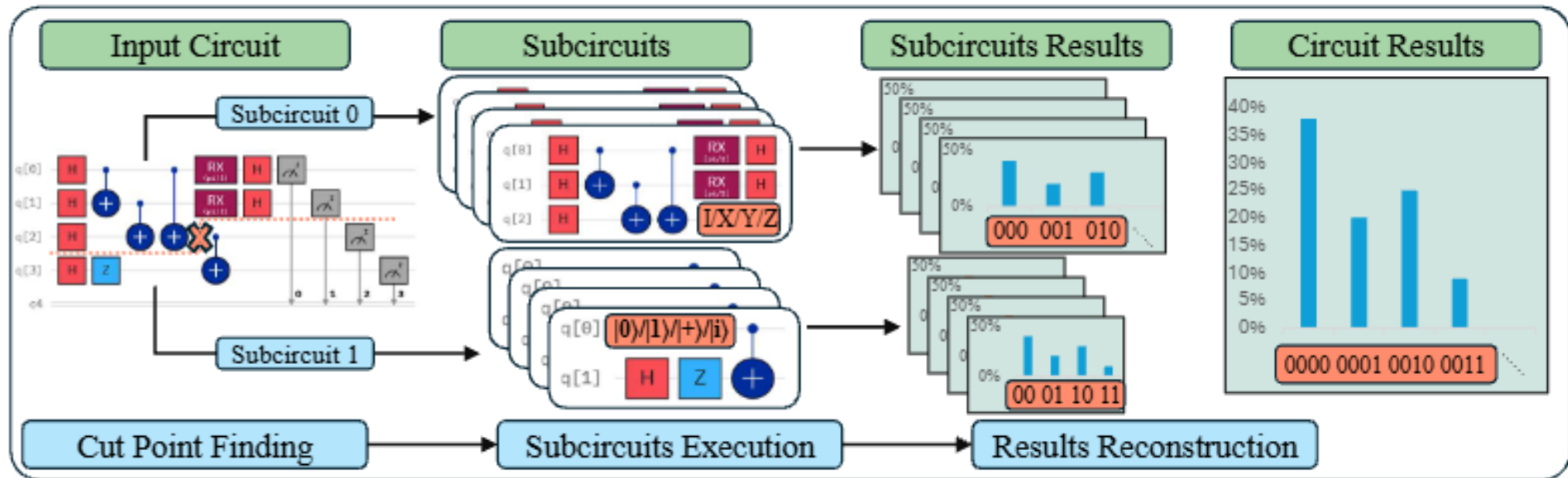
Link quality

noisy

Retry overhead

SR dependent

Circuit cutting reduces qubit demand but explodes reconstruction



Every cut creates more fragment executions and reconstruction work.

Classical cost

$O(4^K)$

Quantum cost

$SR^{-e} + \text{noise}$

Balances classical sampling overhead against quantum communication noise.

Two primitives, one joint optimization problem

1

Circuit cutting

Partitions large circuits into executable fragments; every retained boundary multiplies classical reconstruction cost.

2

EPR merging

Uses remote entanglement to remove selected cut boundaries from reconstruction.

3

The trade-off

More EPR links reduce sampling overhead but add link noise, retries, and communication-qubit demand.

4

ECLIPSE

Jointly chooses merges, physical links, and worker placement using circuit structure and hardware noise.

PART 2

ECLIPSE: Design & Cost Model

ECLIPSE coordinates three decisions



Circuit Analyst

Ranks adjacent fragment pairs by the reconstruction savings from an EPR merge.

Selects high-value, non-overlapping merges instead of maximizing link count.



System Reconfiguration

Maps each logical EPR connection onto a low-noise physical worker pair.

Accounts for link success rate and heterogeneous worker quality.

Prevents avoidable communication noise before scheduling.



Scheduler

Places each fragment where its cumulative gate and idle error should be lowest.

Coordinates computation placement with the chosen EPR links.



Input
Circuit



Analyze
Cuts



Map
Links

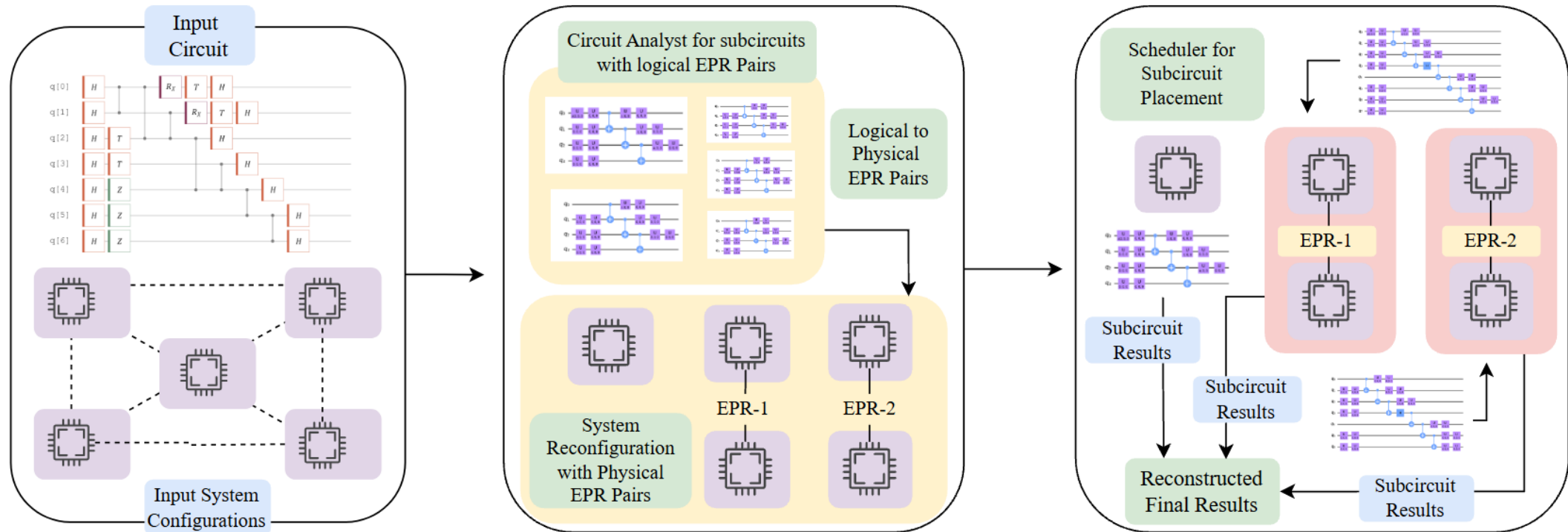


Place
Fragments



Execute +
Rebuild

The complete pipeline jointly optimizes cuts, links, and placement



Paper Figure 4: input circuit + hardware $\rightarrow$ EPR selection and mapping $\rightarrow$ scheduling and reconstruction

One end-to-end plan

Scheduler places fragments on the least damaging workers

Score each fragment-worker assignment using the worker noise profile and the fragment gate structure.

Assign the most error-sensitive fragments to quieter workers while respecting chosen EPR endpoints.

Greedy update the remaining choices; the method is practical rather than globally exact.

Execute fragments, then combine outputs through the reduced reconstruction problem.

The full pipeline is noise-aware from cut selection through physical execution.

PART 3

Evaluation & Results

Real IBM calibration data

ECLIPSE improves fidelity

Compared with Qiskit-Addon-Cut and random EPR placement on real hardware

16.7 %

higher multi-node
fidelity (up to)

99.5 %

overhead reduction
at 8 workers

3 modules

co-optimize cuts,
links, and placement

Workloads — **BV**
ADDER
HWEA
Supremacy

Metrics — **Fidelity**
cumulative
overhead

Baseline — **Qiskit-Addon-Cut**
Addon-Cut-Random-X
ECLIPSE

Random placement is consistently worse: link quality and worker assignment determine whether EPR merging pays off.

PART 4

Discussion & Takeaways

Why it works, what's next

Discussion & Takeaways

1 | SYSTEM INSIGHT

Jointly balance $O(4^K)$ reconstruction and EPR noise.

2 | METHOD

Select merges, map links, and schedule subcircuits.

3 | SCOPE

System-level SR model; greedy algorithms.

4 | RESULTS

+16.7% fidelity; +40.8% deep circuits; -99.5% overhead.

5 | Next

Hardware software co-design compiler

Thank You — Questions?