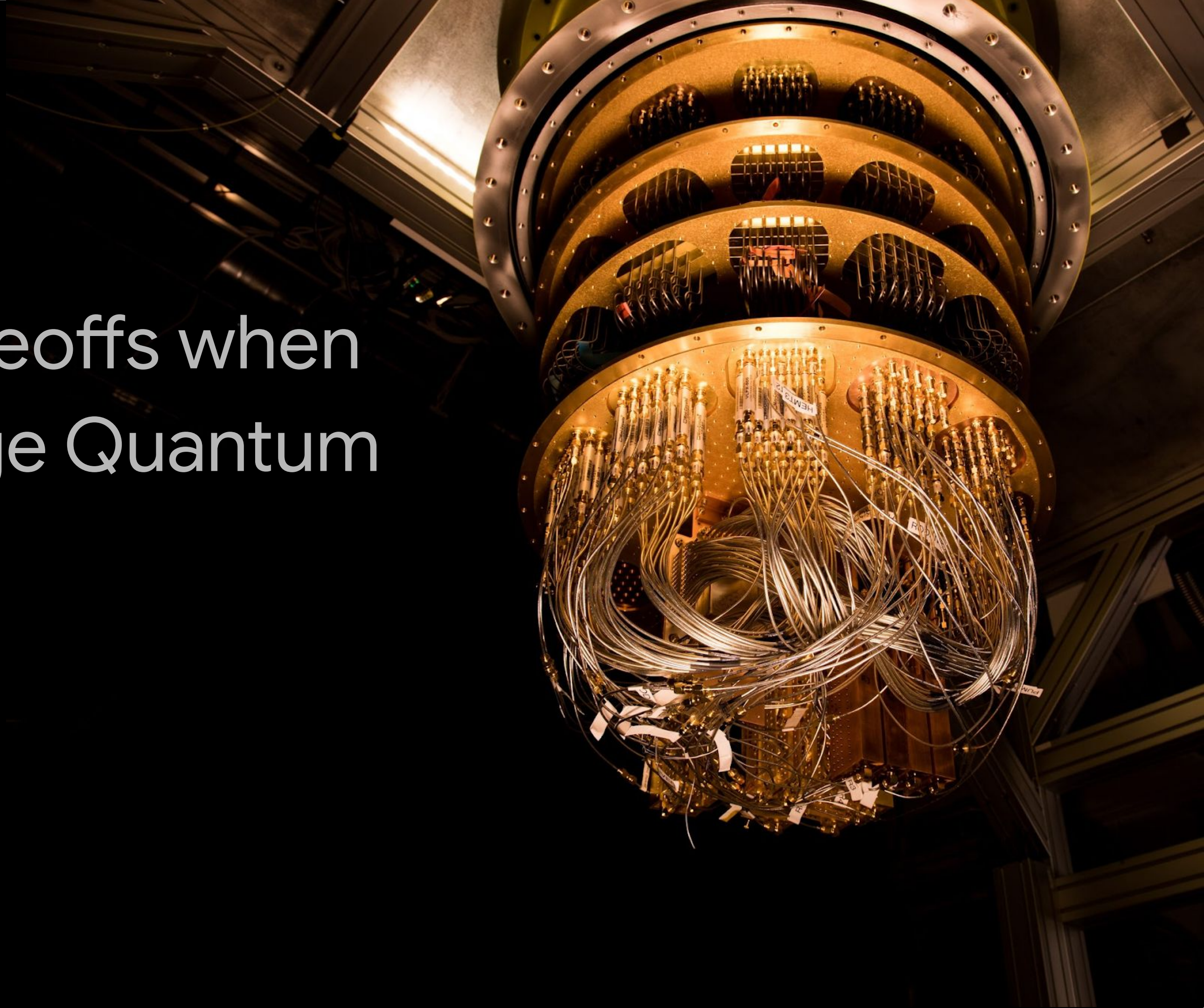
Spacetime Tradeoffs when Optimizing Large Quantum Computations

Craig Gidney



Google AI
Quantum

Outline

1. Gate teleportation and Pauli frames

2. Magic state distillation

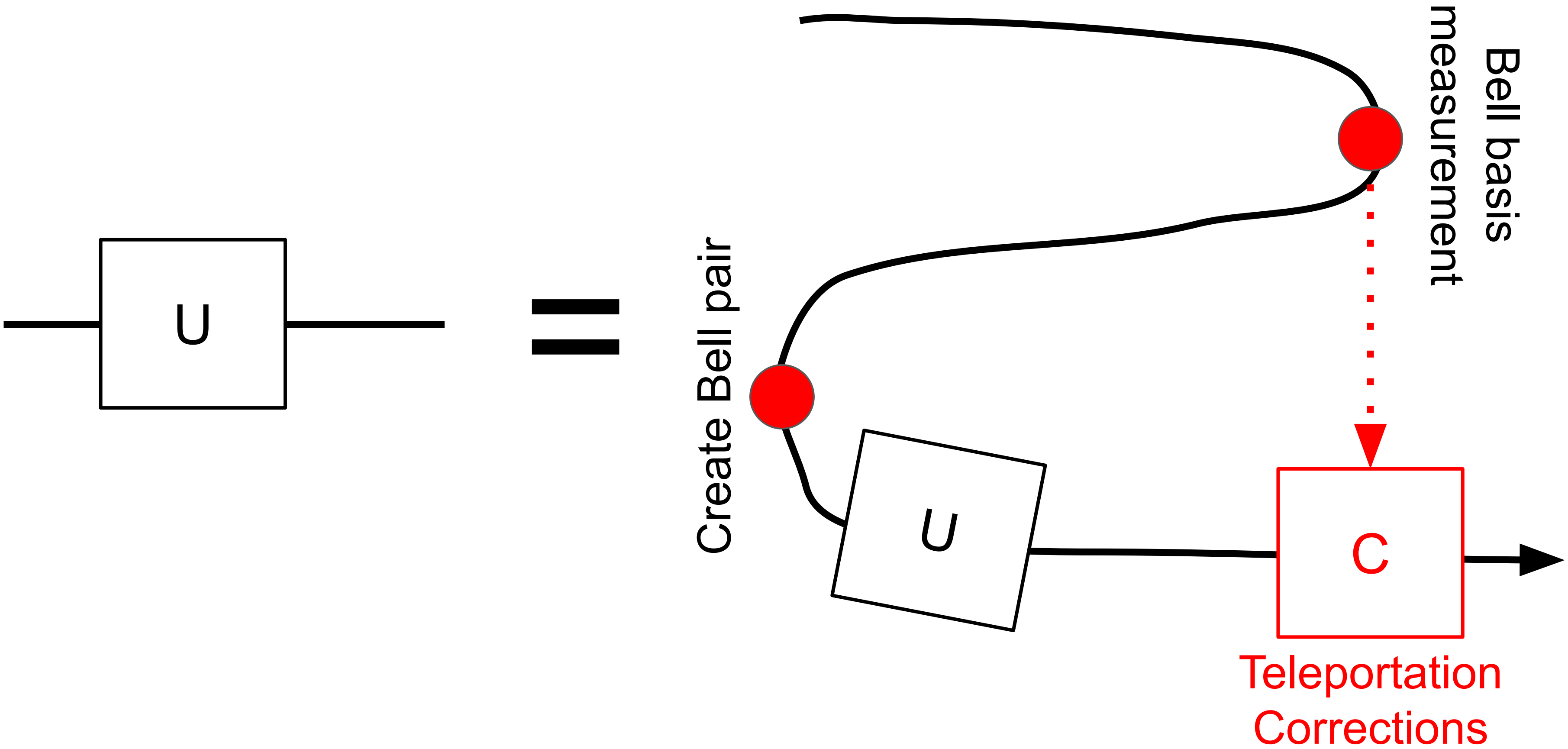
3. Reaction limited computation

4. Reaction limited addition

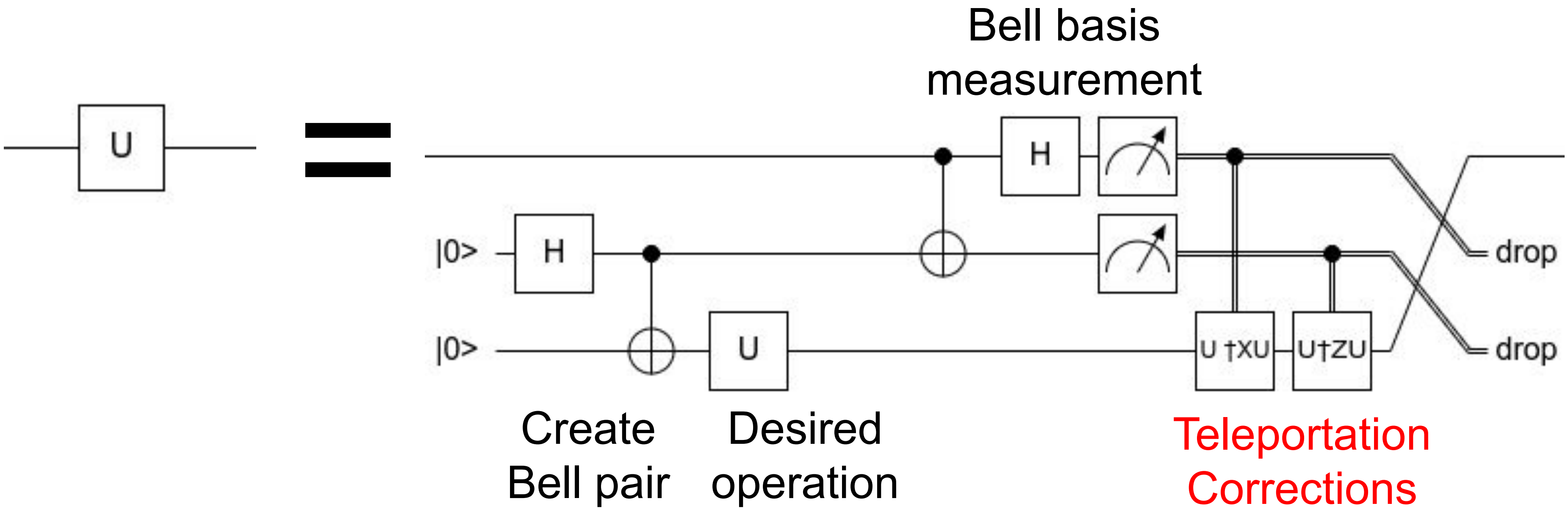
5. Tradeoffs

Part 1: Gate Teleportation and Pauli Frames

Gate teleportation



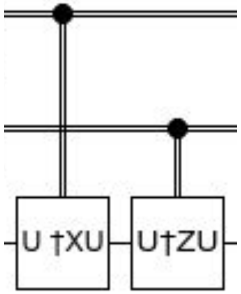
Gate teleportation circuit



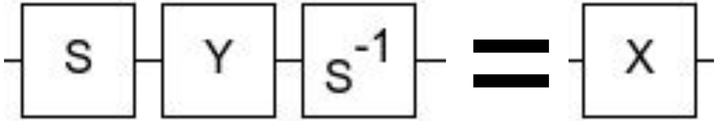
Clifford operations have Pauli corrections

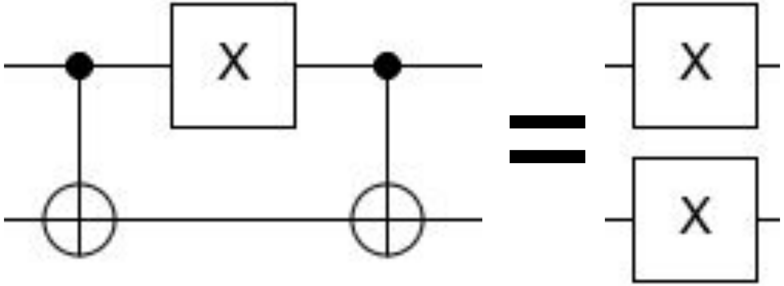
Pauli: X, Y, or Z
Clifford: H, S, CNOT, CZ, etc

Clifford†·Paulis·Clifford = Paulis
(by definition)

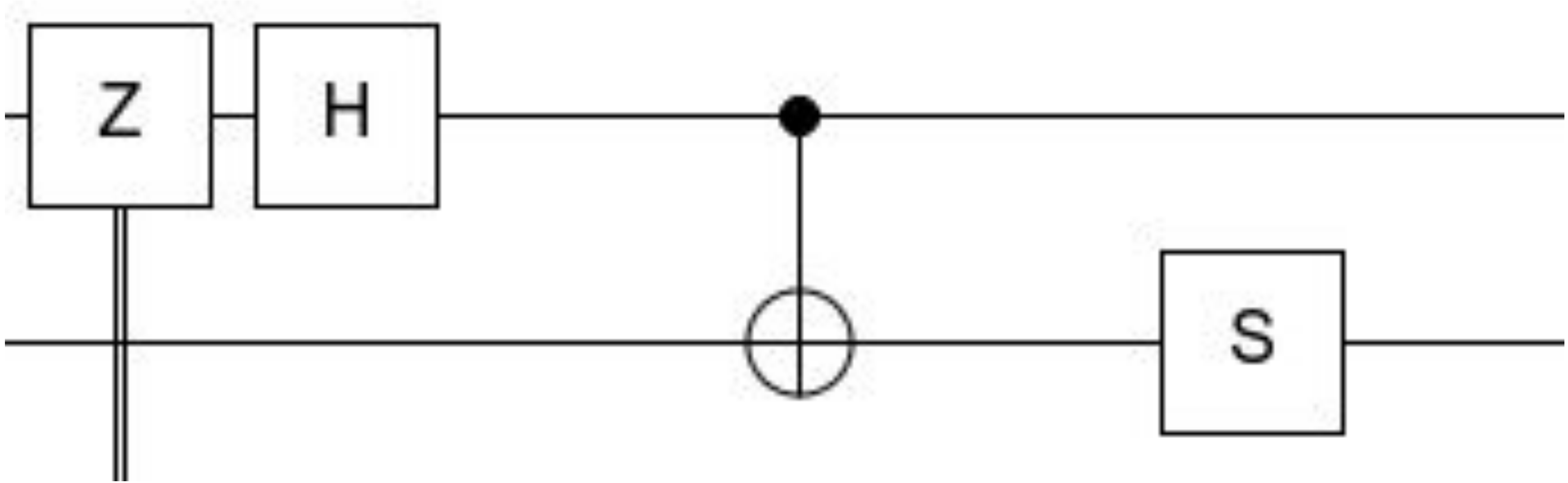


Teleportation Corrections

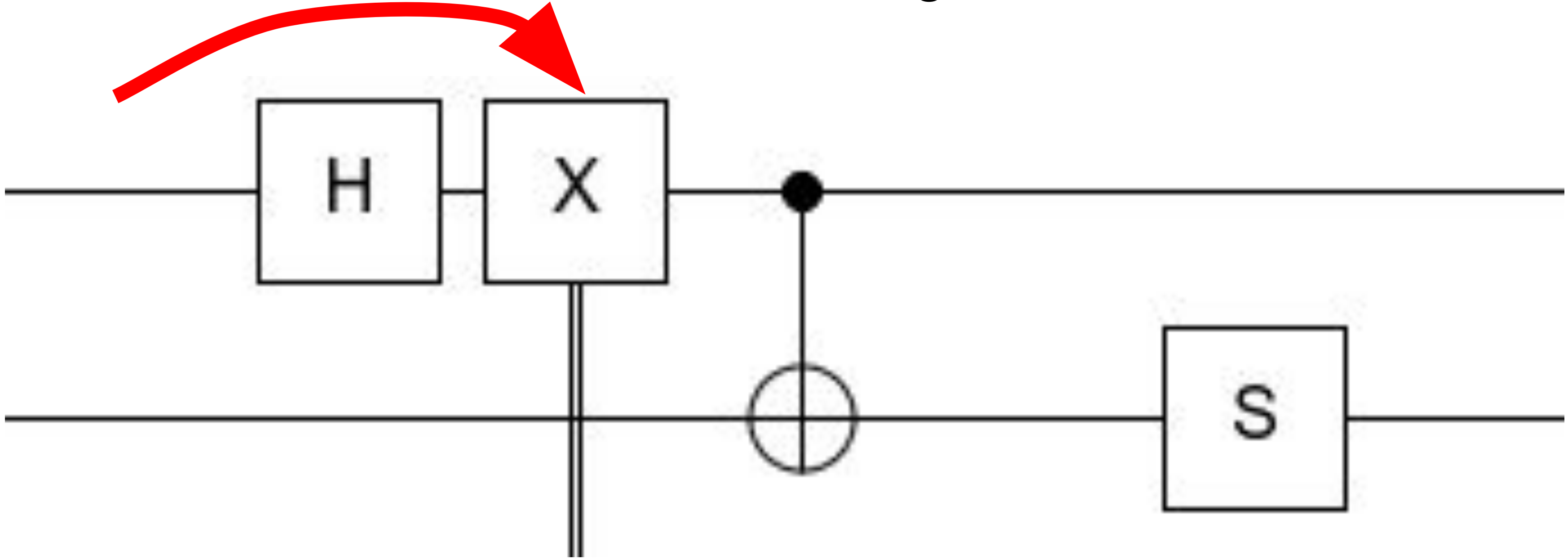




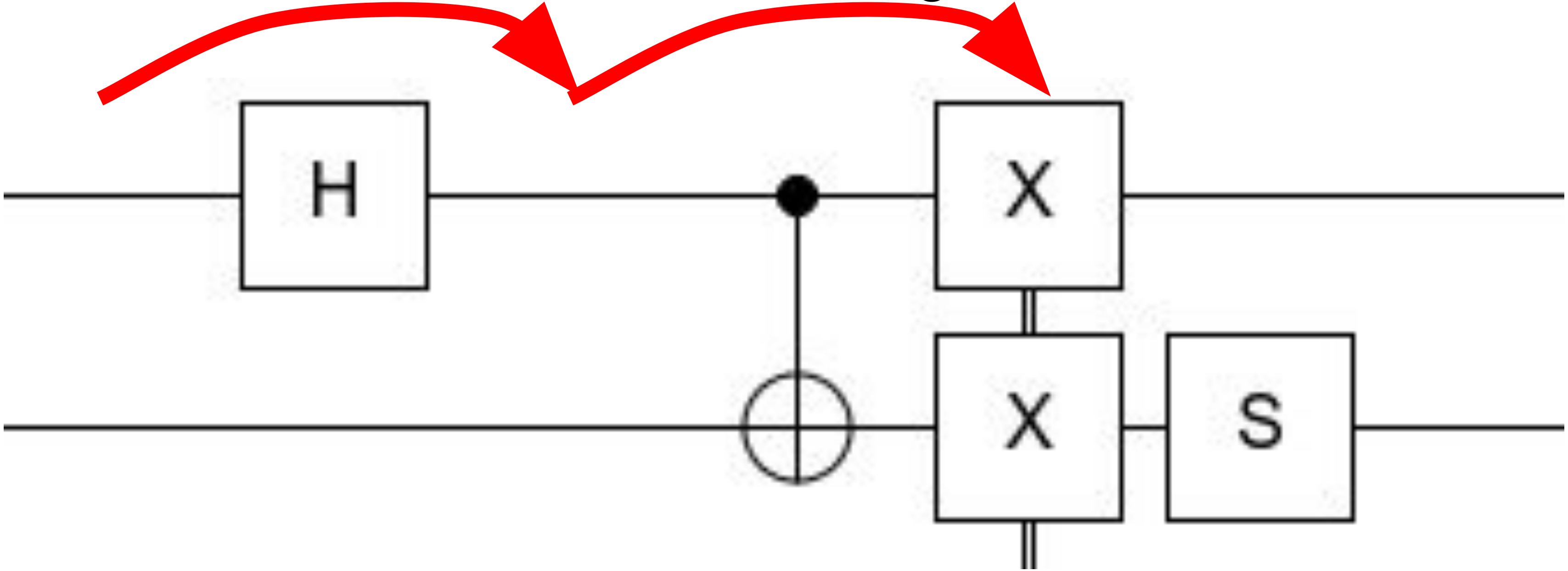
Pauli corrections can move through Clifford circuits



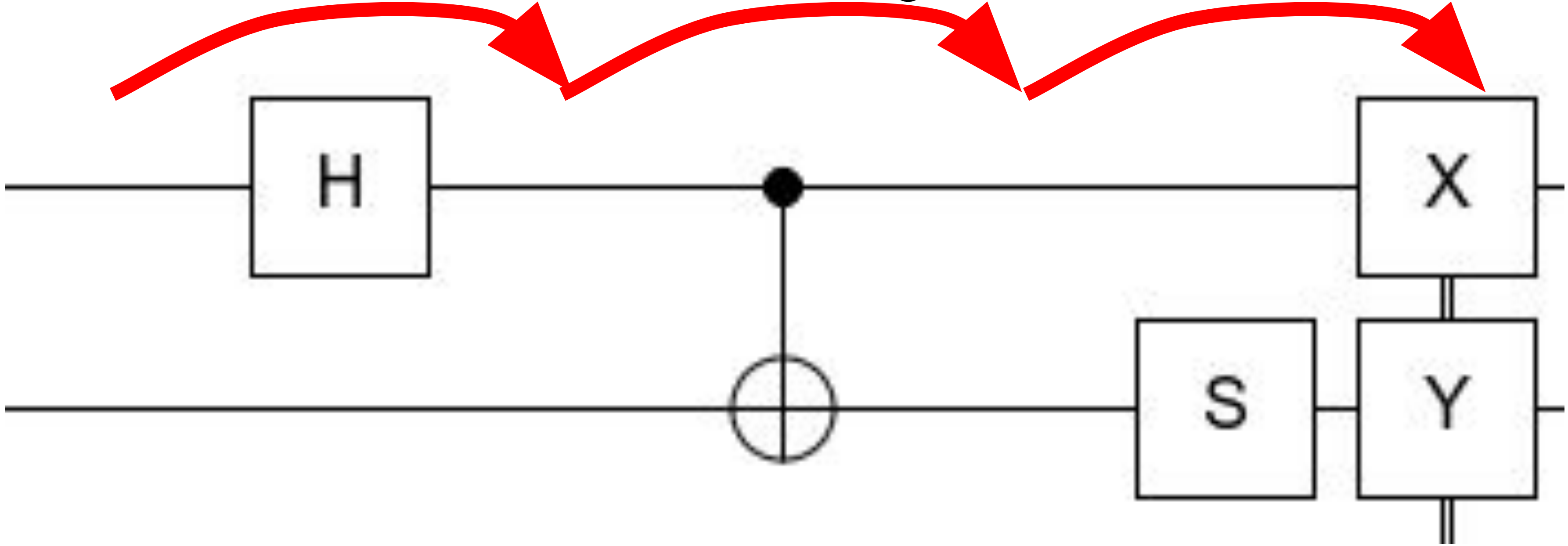
Pauli corrections can move through Clifford circuits



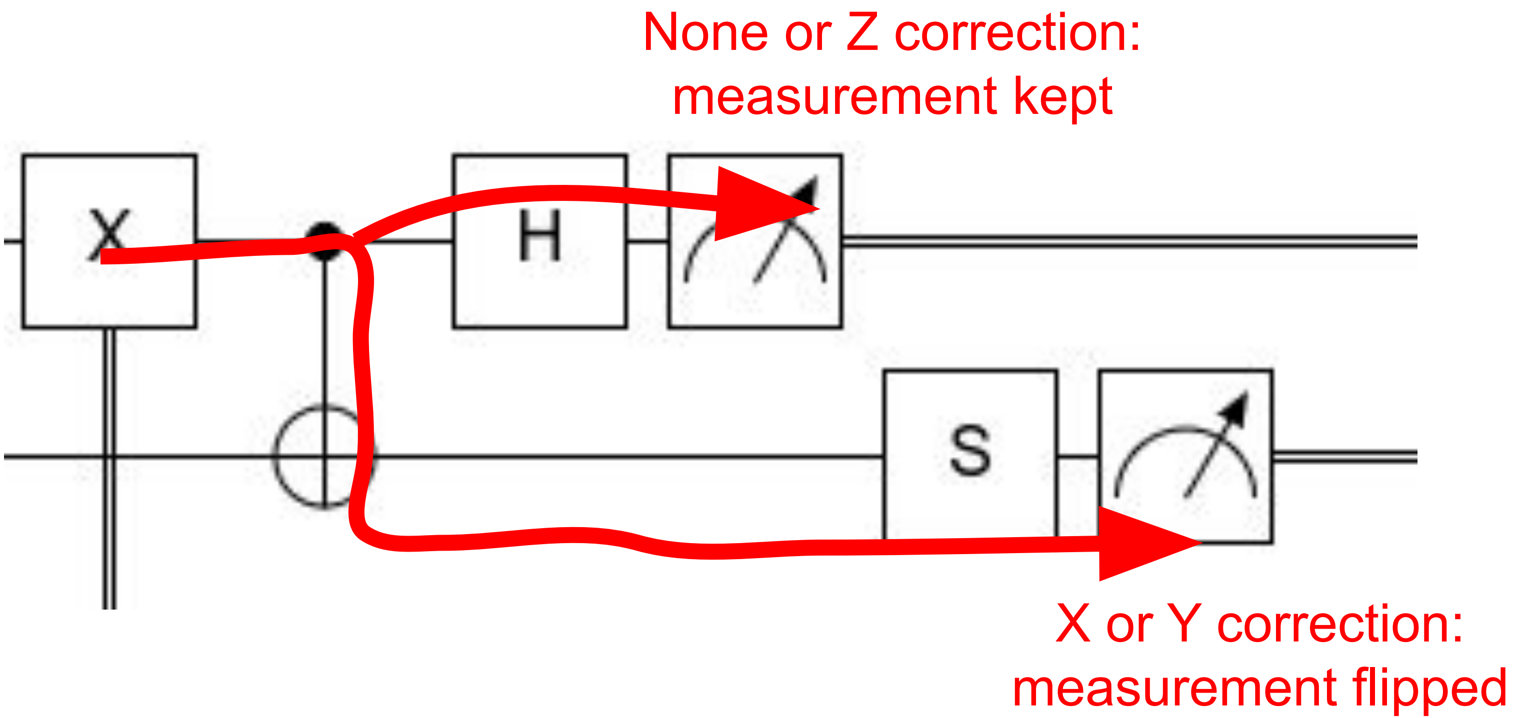
Pauli corrections can move through Clifford circuits



Pauli corrections can move through Clifford circuits



Pauli corrections ultimately only flip measurements



Tactic: Track Pauli corrections ***classically***

- Make entire circuit out of Clifford gates

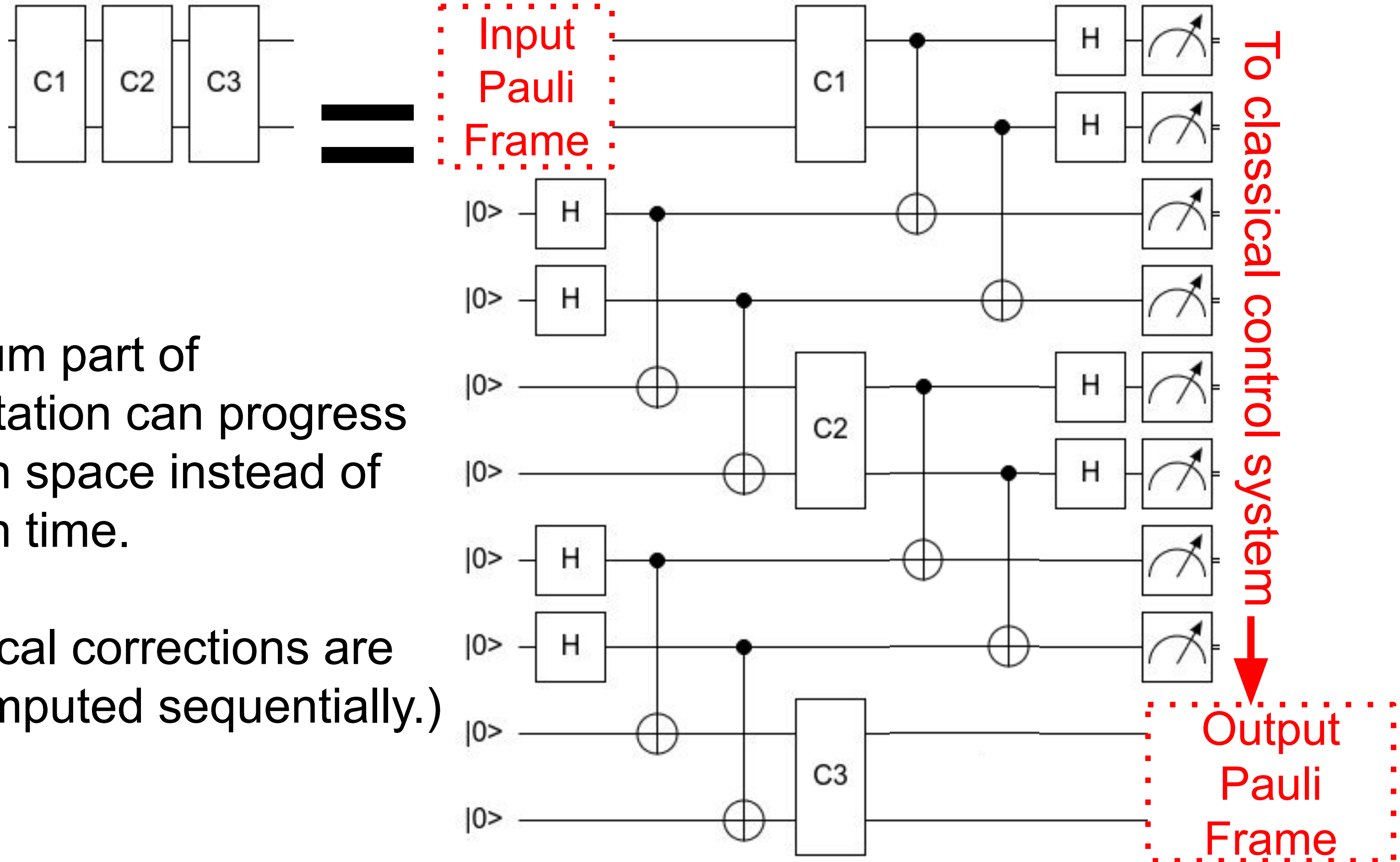
- Teleport as much as you want

- Don't ever apply Pauli corrections to qubits

- Instead, track corrections in the classical control system
  This is called keeping a *Pauli frame*

- Determine which measurements the Pauli frame flipped

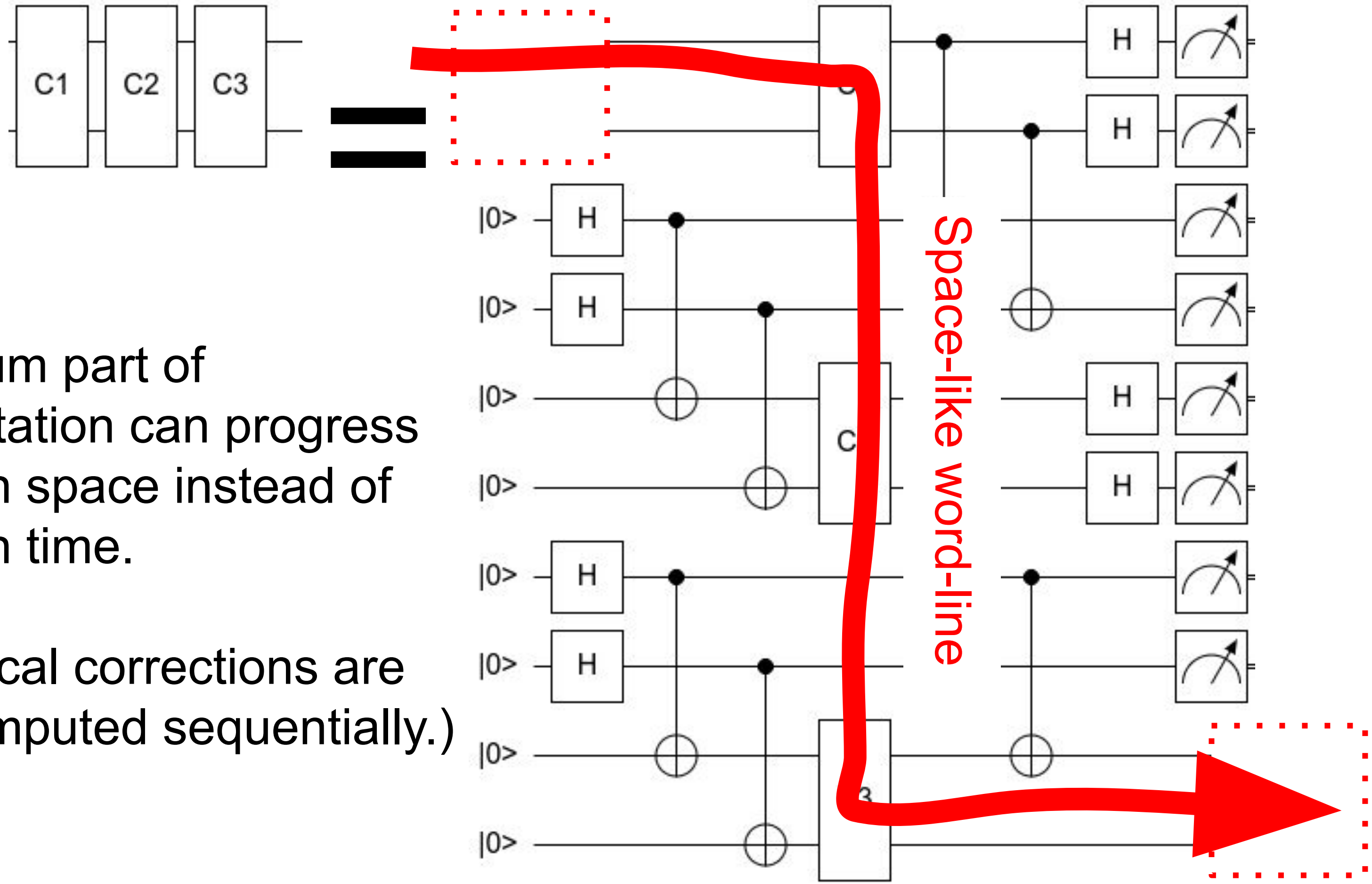
Consequence: Clifford circuits can use space like time



Quantum part of computation can progress through space instead of through time.

(Classical corrections are still computed sequentially.)

Consequence: Clifford circuits can use space like time



Quantum part of computation can progress through space instead of through time.

(Classical corrections are still computed sequentially.)

Part 2: Magic State Distillation

Foreign gates

Full quantum computation requires more than just Clifford gates

(e.g. Clifford circuits can be efficiently simulated classically)

The error correcting code we want to use only natively protects Clifford gates

How do we get non-Clifford functionality out of a Clifford-only code?

Foreign gates

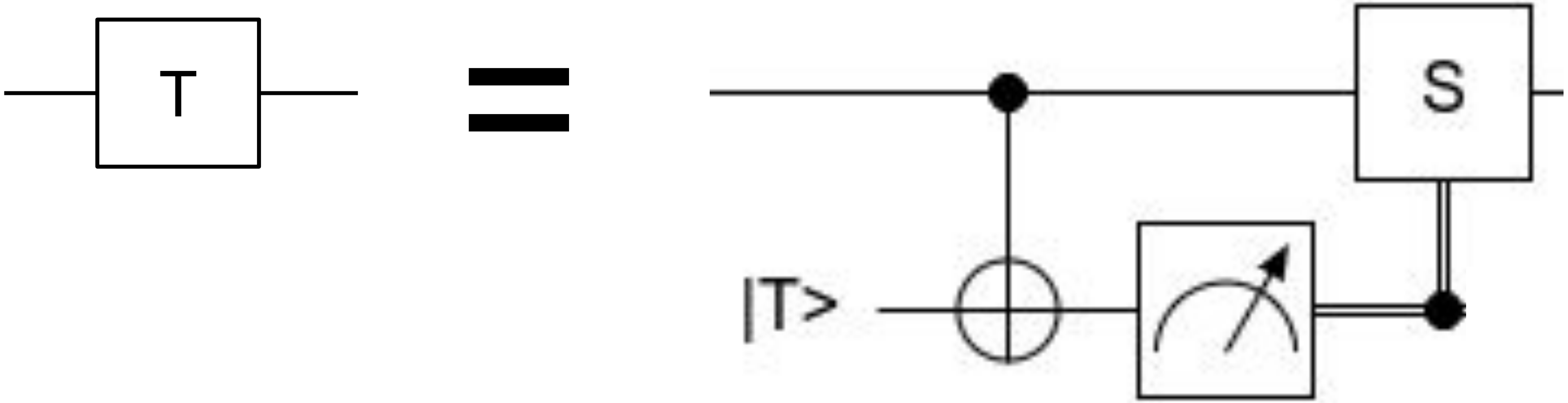
Problem:

- Non-Clifford operations like the T gate are not native to the surface code

Inefficient Solution (code switching):

- Use a T-compatible code to create a Bell pair with a T gate applied
- Convert the Bell pair from the other code into the surface code
- Teleport a target qubit through the Bell pair

T gate from magic T state |T> = T H |0>



Foreign gates

Problem:

- Non-Clifford operations like the T gate are not native to the surface code

Less Inefficient Solution (**magic state distillation**):

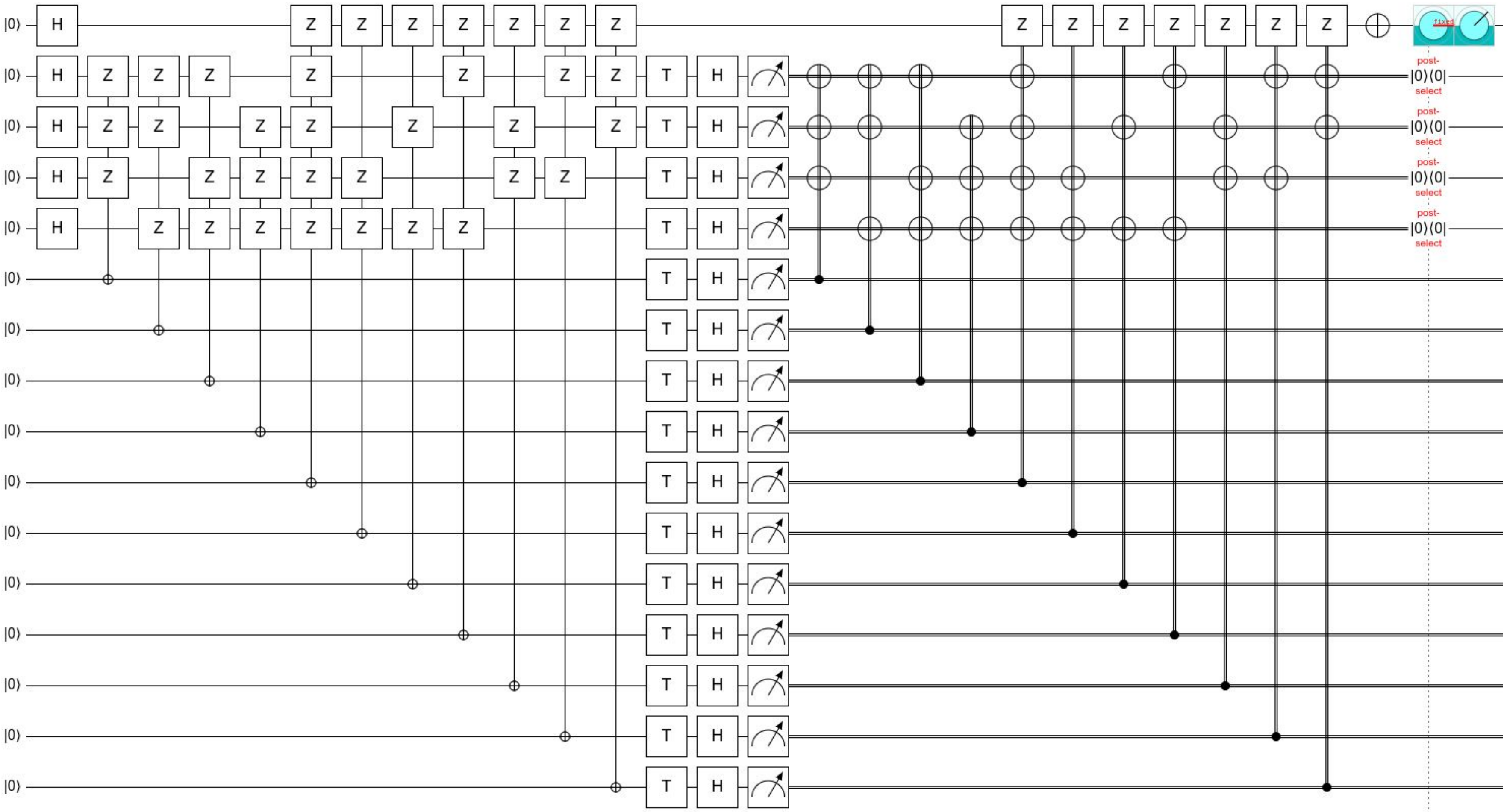
- Prepare noisy T|+> states directly in the surface code

- Run destructive cross-checks on the T states to detect errors

- Apply T gates by teleporting through surviving T states

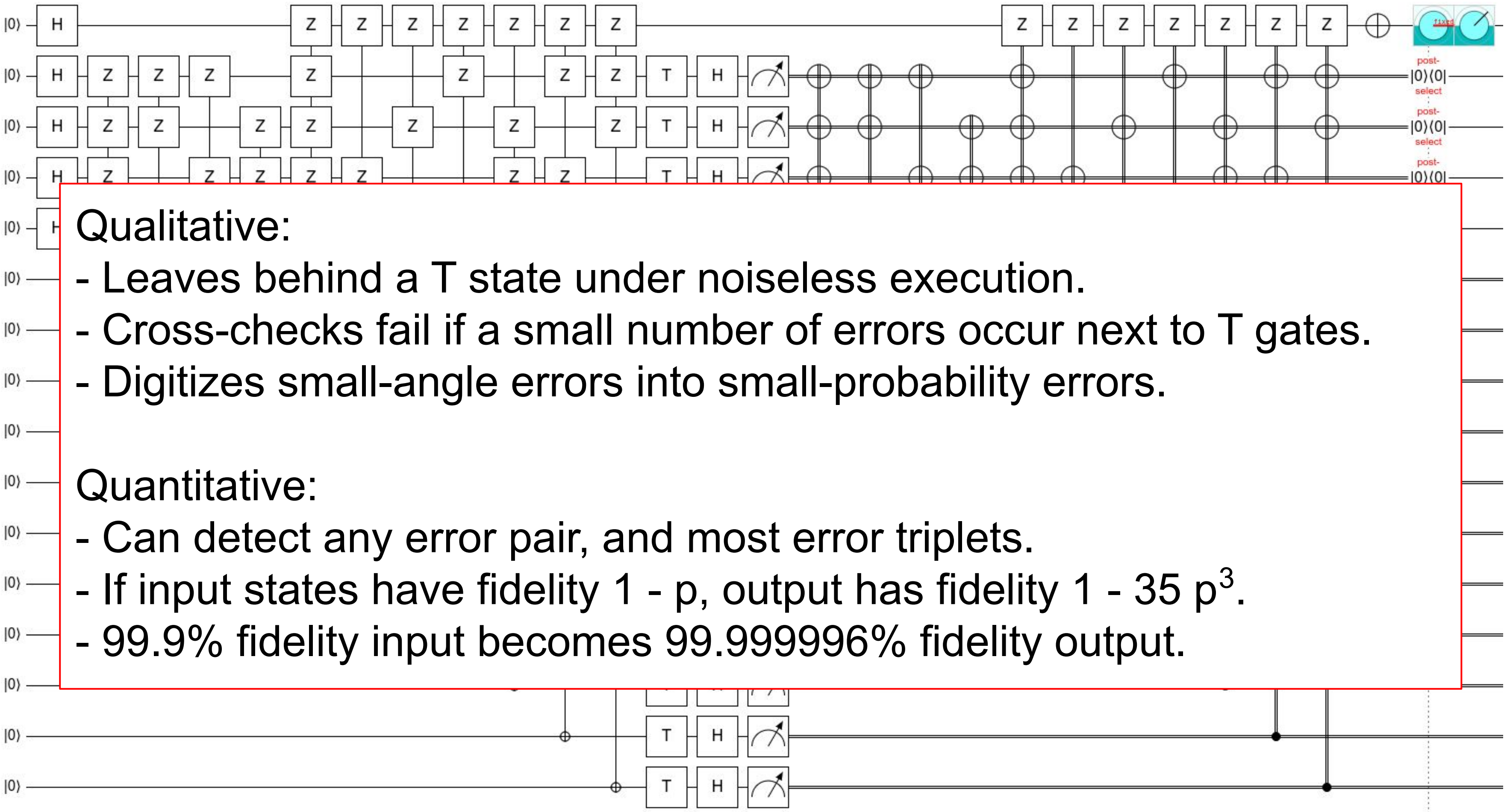
15-to-1 T state distillation

Figure source: example circuit at algassert.com/quirk



15-to-1 T state distillation

Figure source: example circuit at algassert.com/quirk

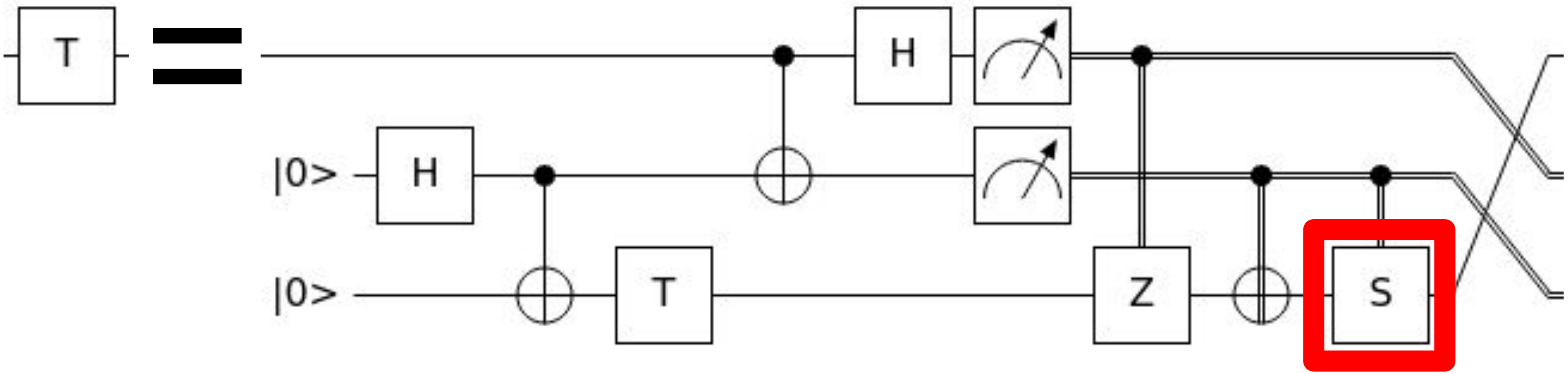


Qualitative:
- Leaves behind a T state under noiseless execution.
- Cross-checks fail if a small number of errors occur next to T gates.
- Digitizes small-angle errors into small-probability errors.

Quantitative:
- Can detect any error pair, and most error triplets.
- If input states have fidelity 1 - p, output has fidelity 1 - 35 p^3.
- 99.9% fidelity input becomes 99.999996% fidelity output.

Part 3: Reaction Limited Computation

Non-Clifford gates have non-Pauli corrections



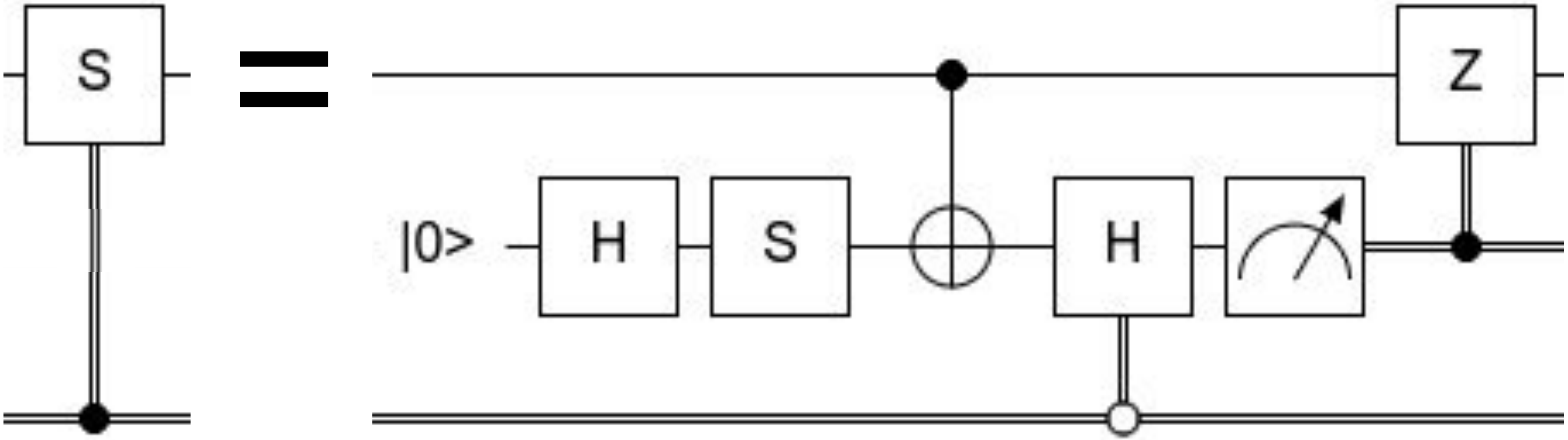
Changing what's conditional

Two problems with the controlled-S gate:

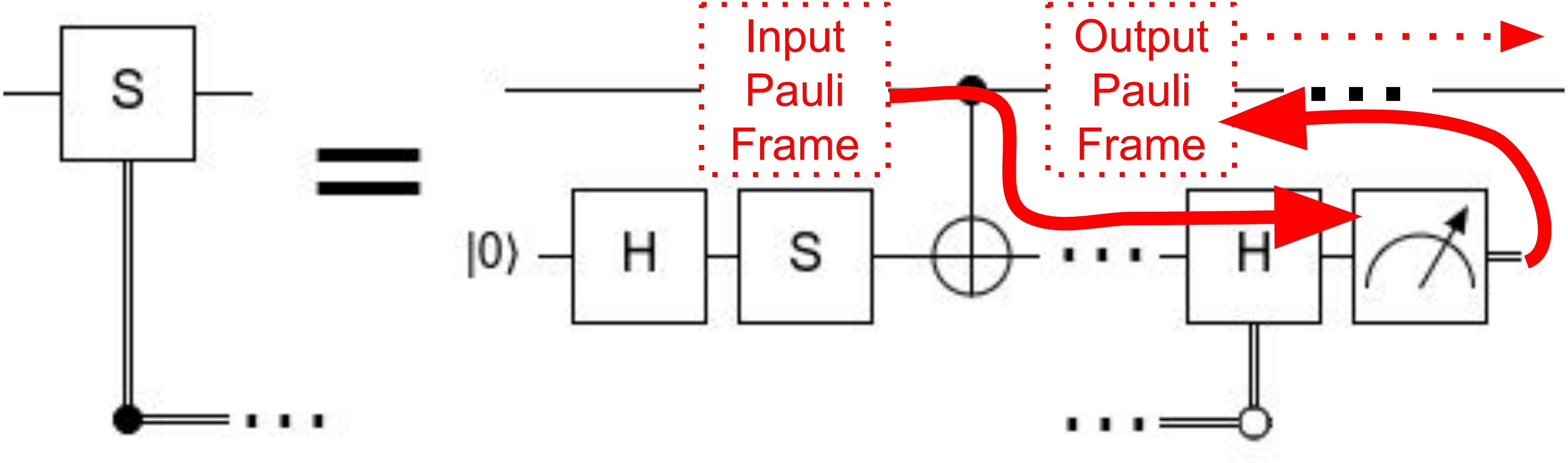
1. It's slow. It's not transversal in the surface code.
2. It's in the way. Delaying more operations on the target.

Fix #1: Move the conditional operation onto a worker qubit.

Fix #2: Change the conditional operation to a transversal operation (H).



Tactic: Delayed Choice Gate

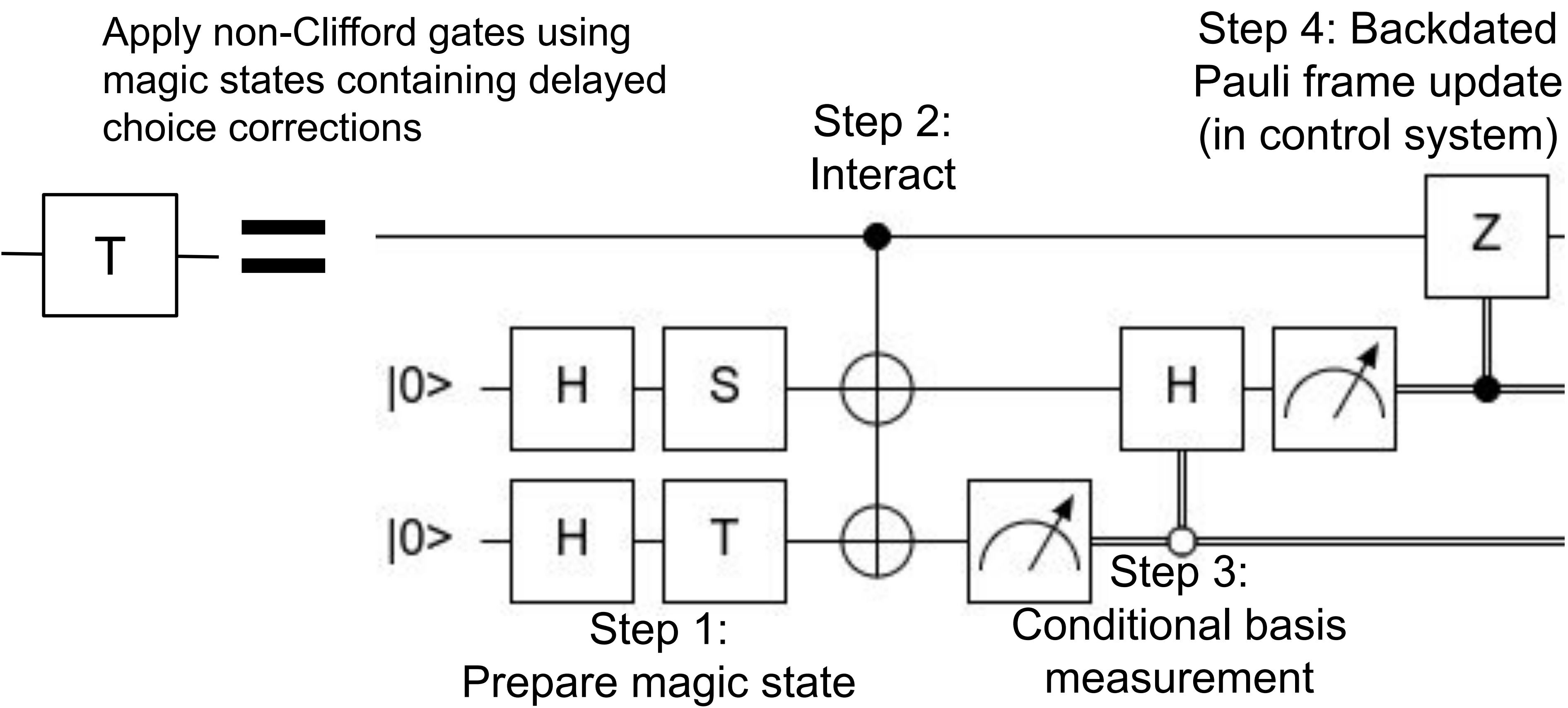


1. The target qubit is interacted with a *routing qubit* storing a magic state.
2. The routing qubit is kept until the choice to apply the S is known.
3. The choice determines the basis to measure the routing qubit in.
4. Pauli frame corrections are computed from the measurement basis and result.

First AutoT: [arXiv:1808.02892](https://arxiv.org/abs/1808.02892)

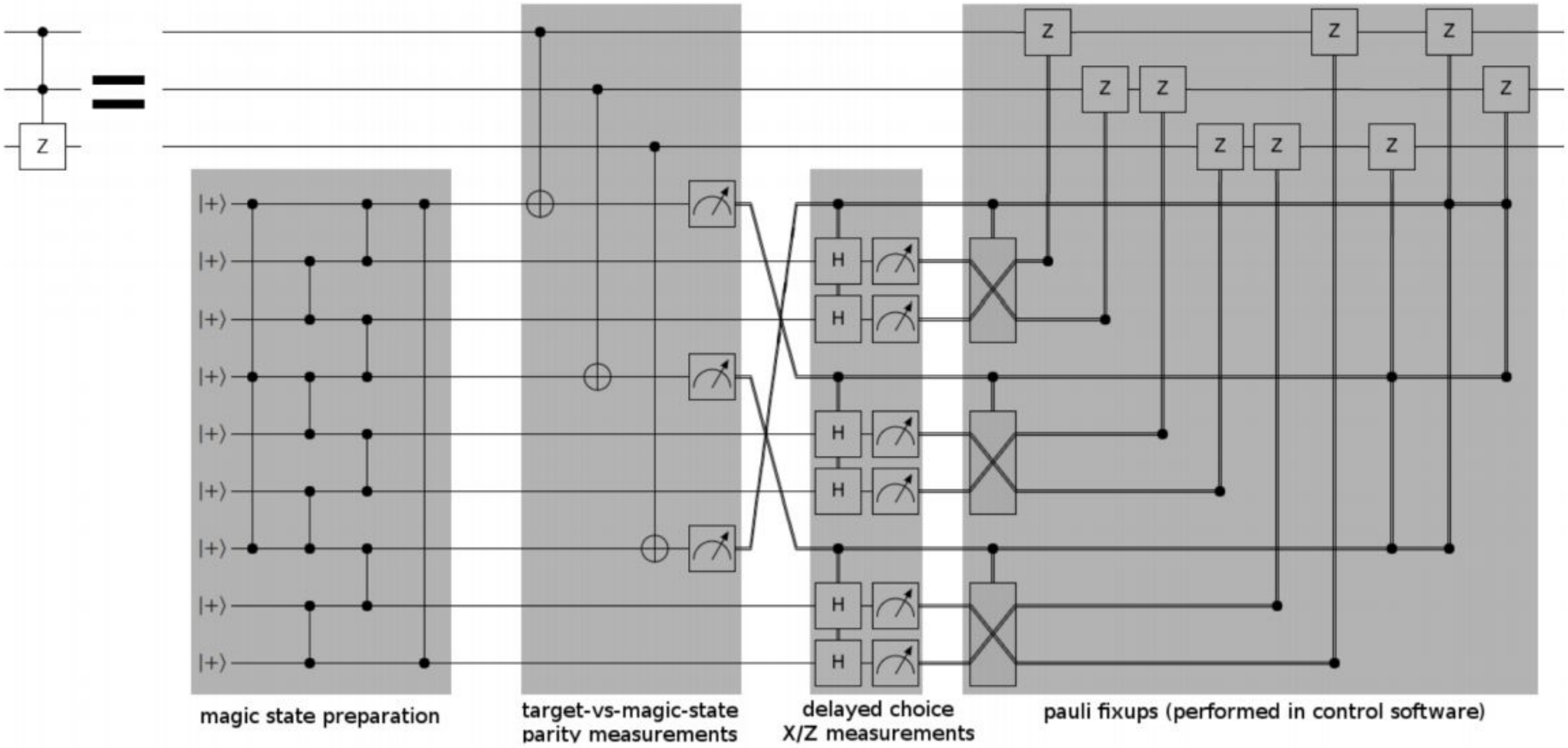
Tactic: Auto-magic states

Apply non-Clifford gates using magic states containing delayed choice corrections

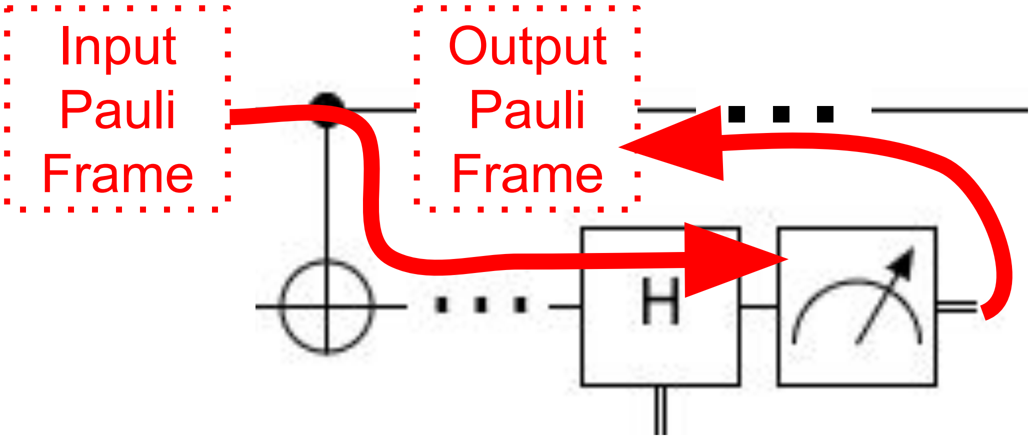


More auto-magic: an AutoCCZ circuit

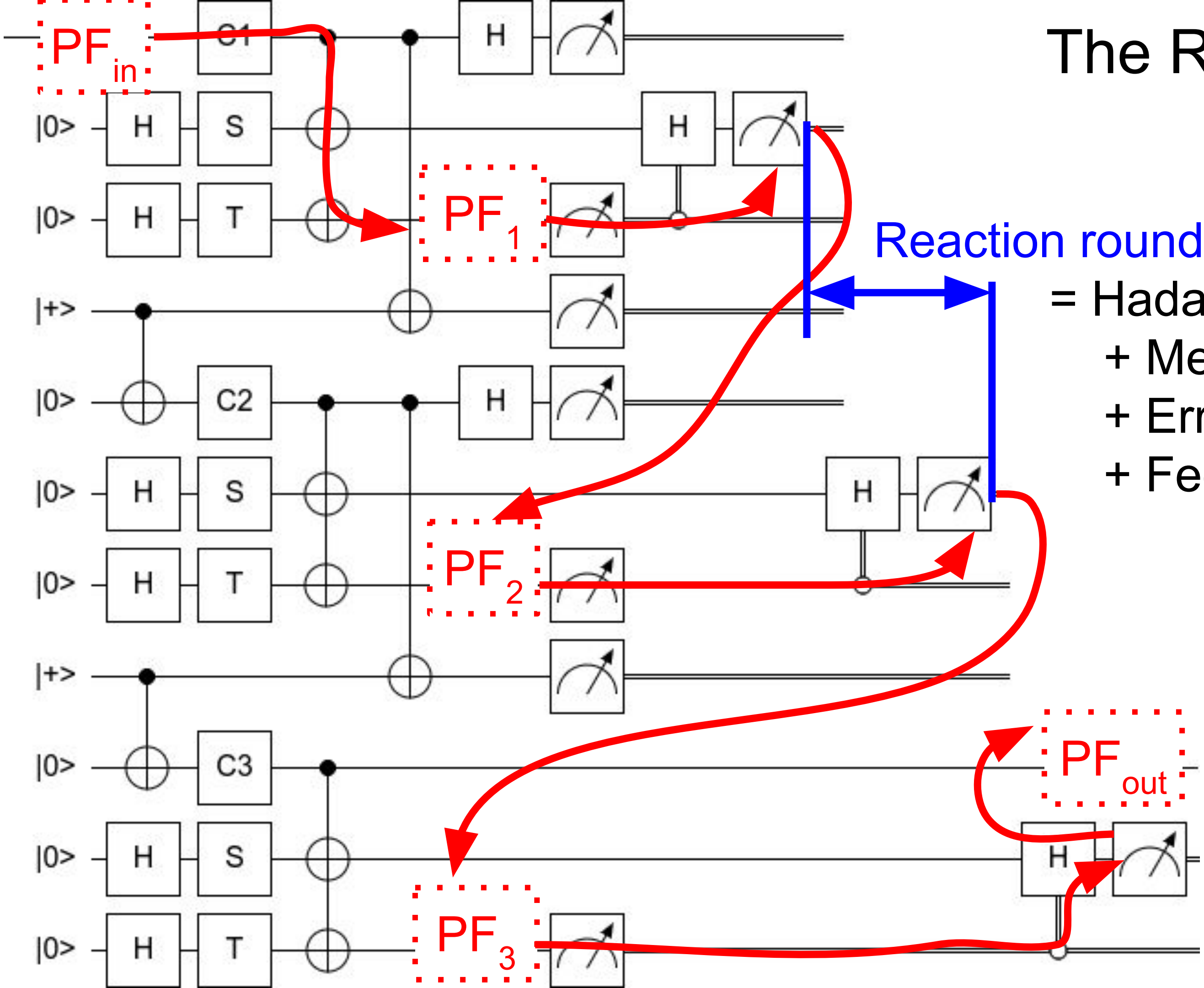
Figure source: [arXiv:1905.08916](https://arxiv.org/abs/1905.08916)



The Reaction Cycle



The Reaction Cycle



= Hadamard duration
+ Measure duration
+ Error correction delay
+ Feedback delay

Tactic: Reaction limited computation

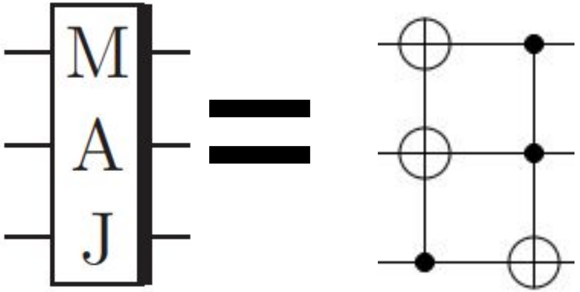
1. Parallelize the computation using copious gate teleportation

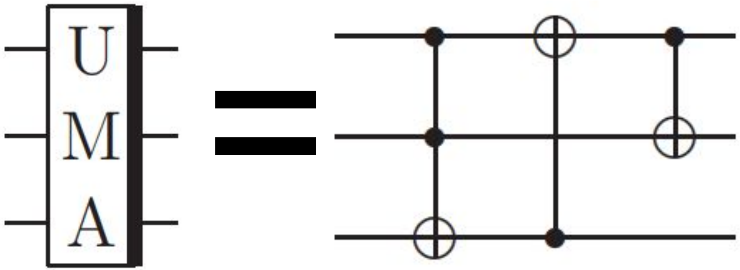
2. Perform non-Clifford operations using auto-magic states, leaving routing qubits

3. Resolve routing qubit chains as fast as the reaction round trip time allows

Part 4: Reaction Limited Addition

Cuccaro Adder





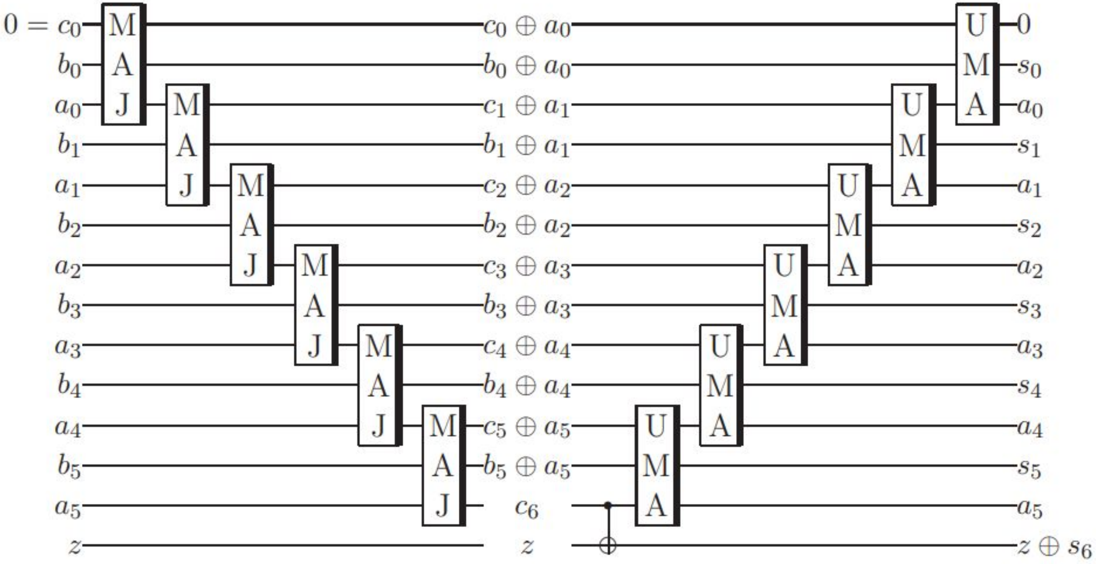
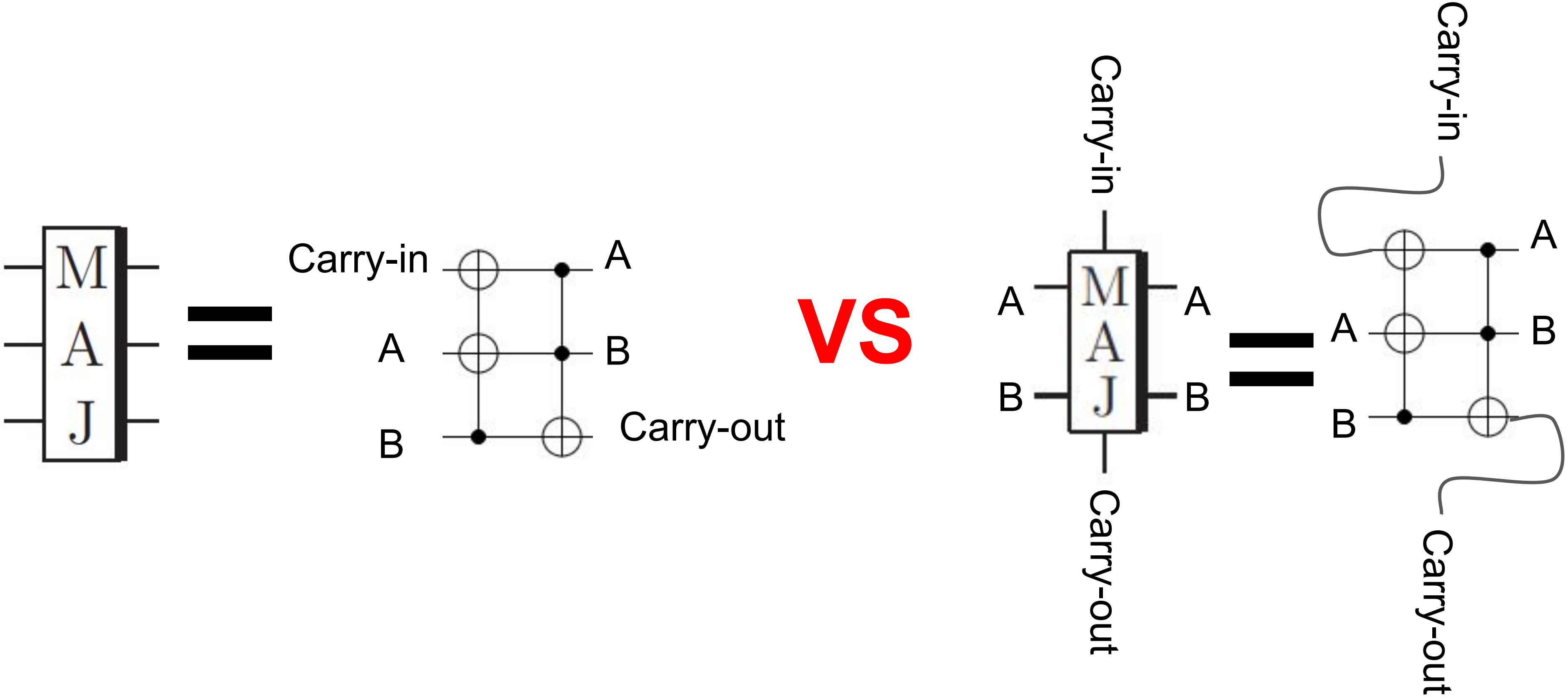
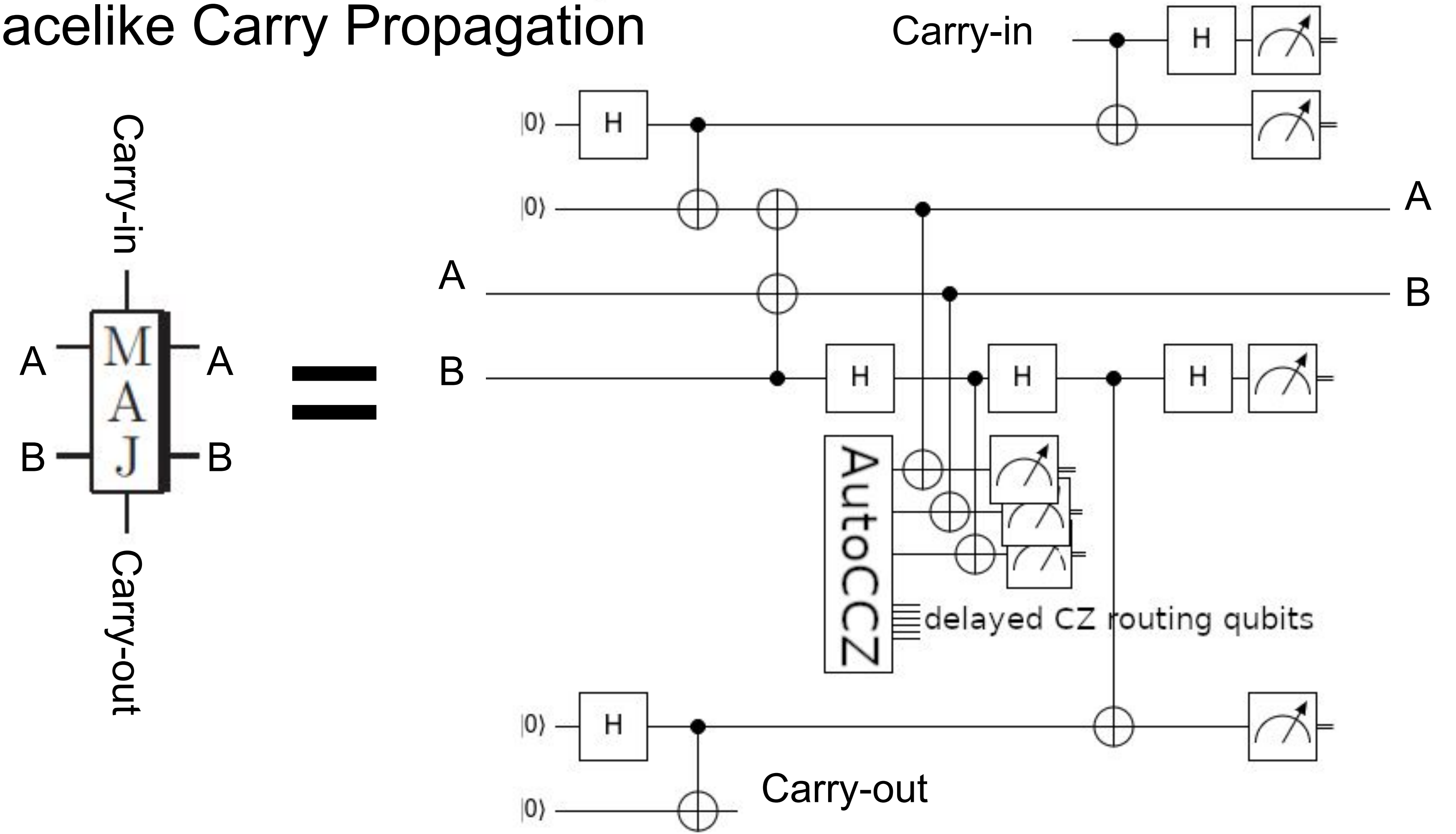


Figure source: arXiv:quant-ph/0410184

Timelike vs Spacelike Carry Propagation



Spacelike Carry Propagation



Cuccaro Adder with Spacelike Layout

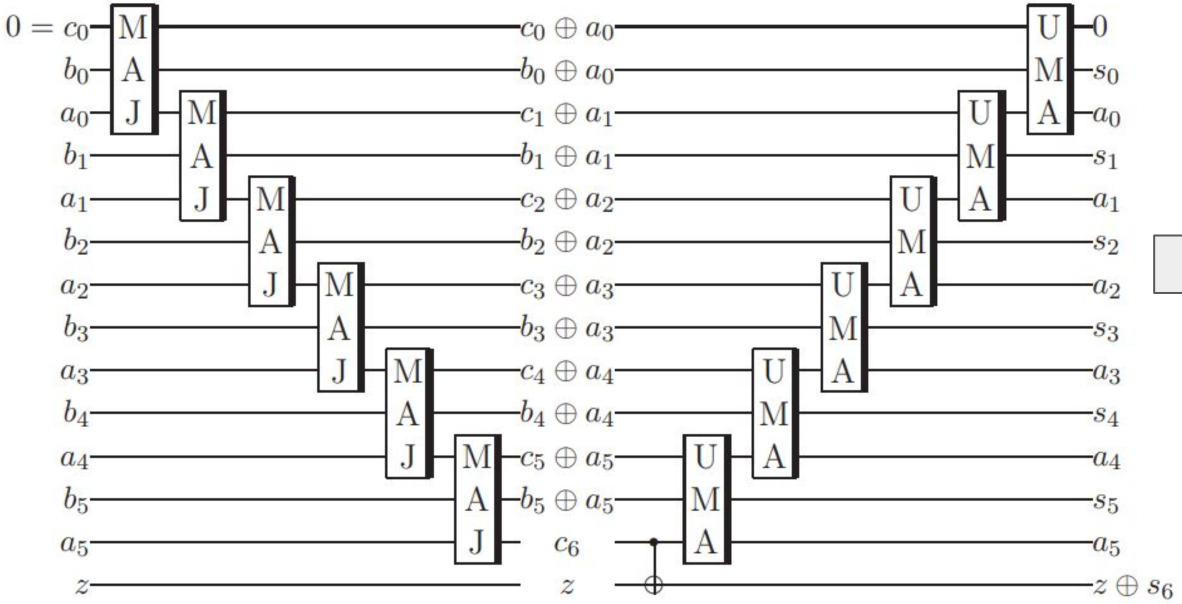
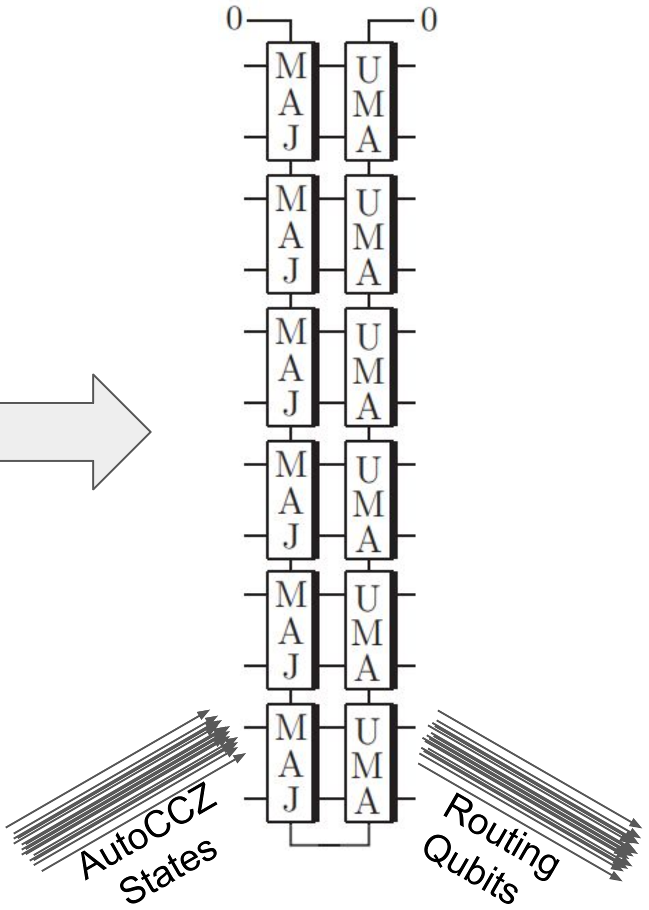


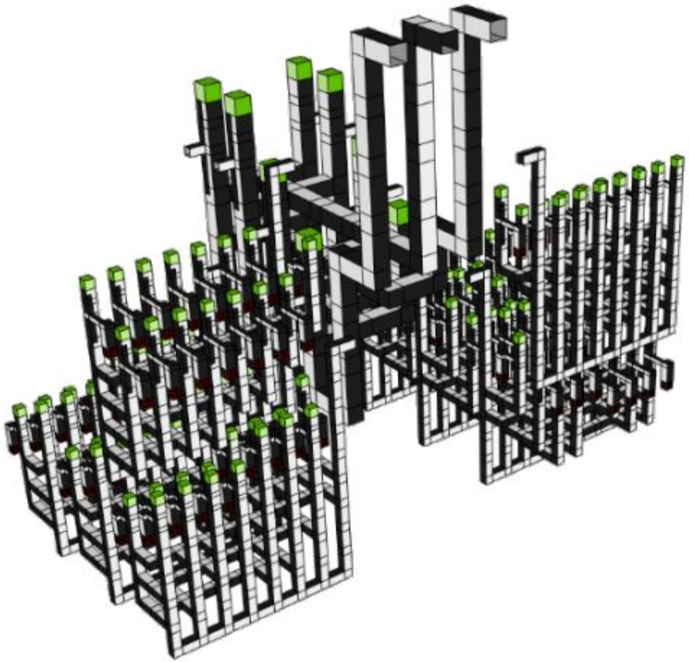
Figure source: arXiv:quant-ph/0410184



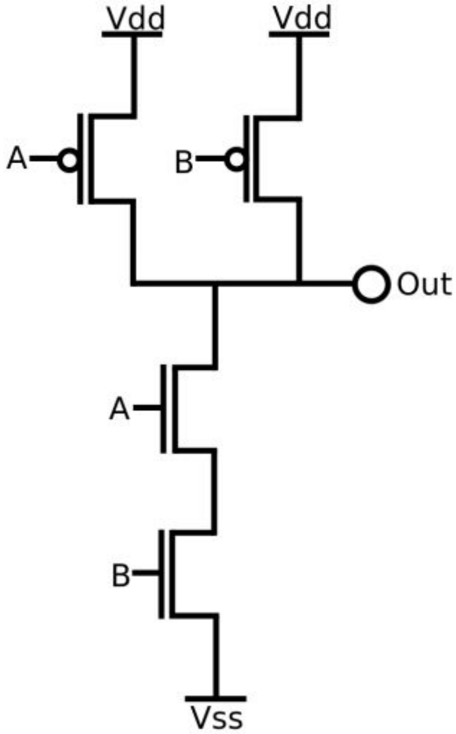
Part 5: Tradeoffs

Context: Quantum fault tolerance is expensive!

Figure source: arXiv:2011.04149



(a) "Quantum NAND"
> 10 qubitseconds



(b) "Classical NAND"
$< 10^{-9}$ transistorseconds

Some rough estimated numbers for scale

Reaction round trip time: 10 microseconds
AutoCCZ factory period: 150 microseconds
AutoCCZ factory footprint: 100 000 physical qubits

It takes 1.5 million physical qubits dedicated to magic state distillation to saturate a single-threaded reaction limited computation.

(keeping in mind overheads are always getting better)

Example: Is a low depth adder worth it?

Reaction round trip time: 10 microseconds
AutoCCZ factory period: 150 microseconds
AutoCCZ factory footprint: 100 000 physical qubits

Toffoli count overhead of using a carry lookahead adder: **5x**
(arXiv:2004.01826 Thapliyal et al)

Physical qubits that must be dedicated to distillation before a low depth adder can finish before a ripple carry adder:
> 15 000 000

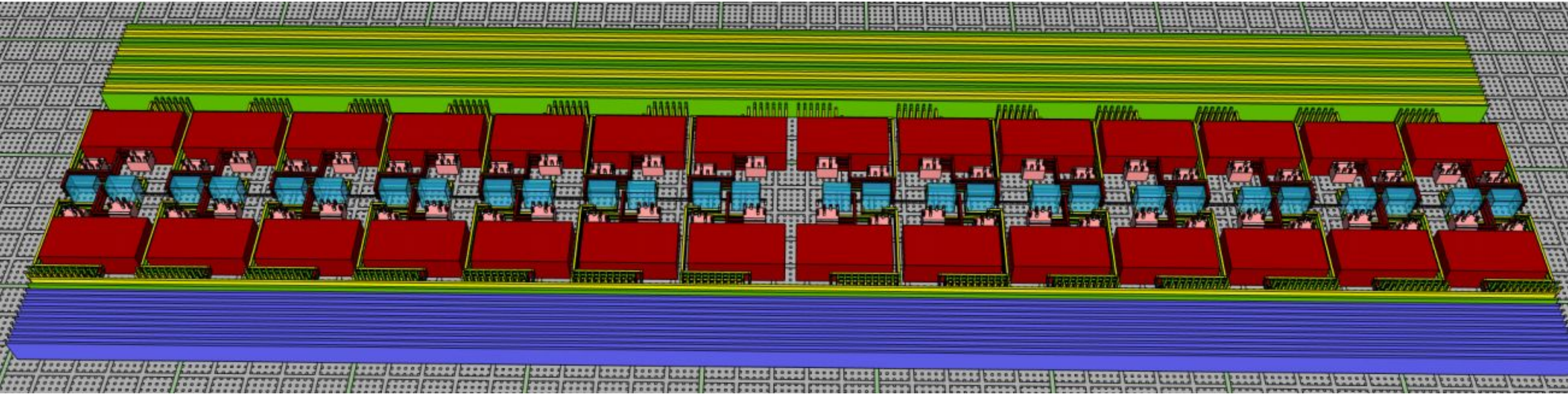
(keeping in mind overheads are always getting better)

Partially spacelike additions speed up factoring

from [arXiv:1905.09749](https://arxiv.org/abs/1905.09749)

Red box = CCZ factory (28 total)
Pink box = AutoCCZ converters
Blue box = MAJ w/ spacelike carry propagation

Green+Yellow rows = interleaved input data



Blue rows = Stored data not being acted on

Left side and right side are separate additions being run in parallel

Review

Review

1. Gate teleportation and Pauli frames

2. Magic state distillation

3. Reaction limited computation

4. Reaction limited addition

5. Tradeoffs

Thanks for listening!

Questions?