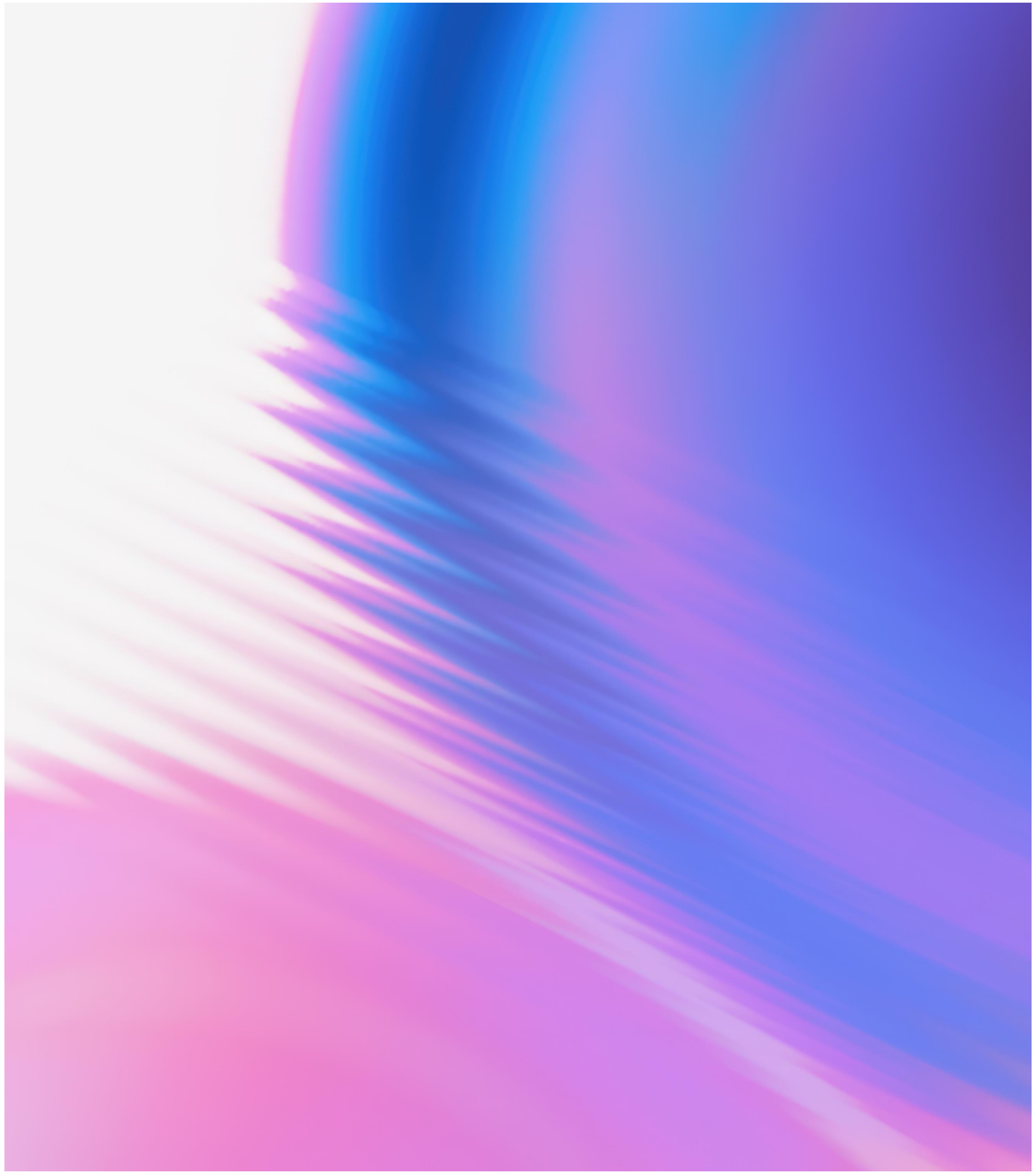
Dynamic Circuits for Efficient Quantum Computation

Elisa ~~Bäumer~~ Marty
Research Scientist
IBM Quantum

Outline:

- IBM Quantum Roadmap
- Dynamic Circuits
- Long-Range Entanglement
- Quantum Fourier Transform
- ➢ *Logarithmic-Depth AQFT on a Line*



IBM

Key milestones in bringing useful quantum computing to the world

2023

Establish quantum utility

2026

Demonstrate quantum advantage

2029

Deliver the first large-scale, fault-tolerant quantum computer

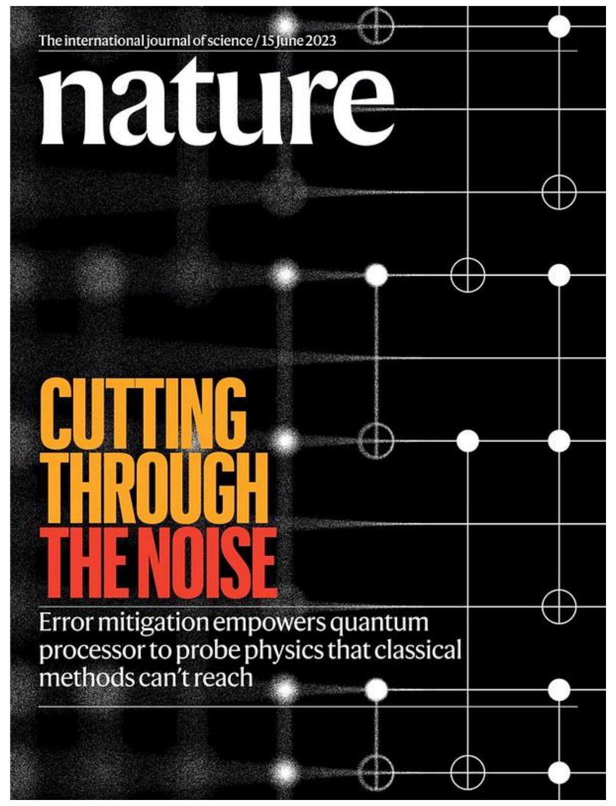
Utility vs. Advantage

Theory Utility Advantage

Quantum Utility (2023)

Demonstration that a quantum computer can run quantum circuits beyond the ability of a classical computer simulating a quantum computer

Confirmation via research, papers, & theory



IBM's 2023 research paper ("Evidence for the utility of quantum computing before fault tolerance") provided evidence and methods to move the industry into the Utility era

https://www.nature.com/articles/s41586-023-06096-3

Quantum Advantage (TBD)

Demonstration that a quantum computer can solve a problem more accurately, cheaper, or more efficiently than classical computing alone

Confirmation via real-world usage

Advantage will come at different times in different domains and depends on the continued advancement of quantum algorithm implementations across industries

What is quantum advantage?

Performing an information processing task more efficiently, cost-effectively, or accurately using quantum computers than is known to be possible with classical computers alone

There are two important nuances to this →

1
We must establish trust in the outputs of a real and noisy quantum device.

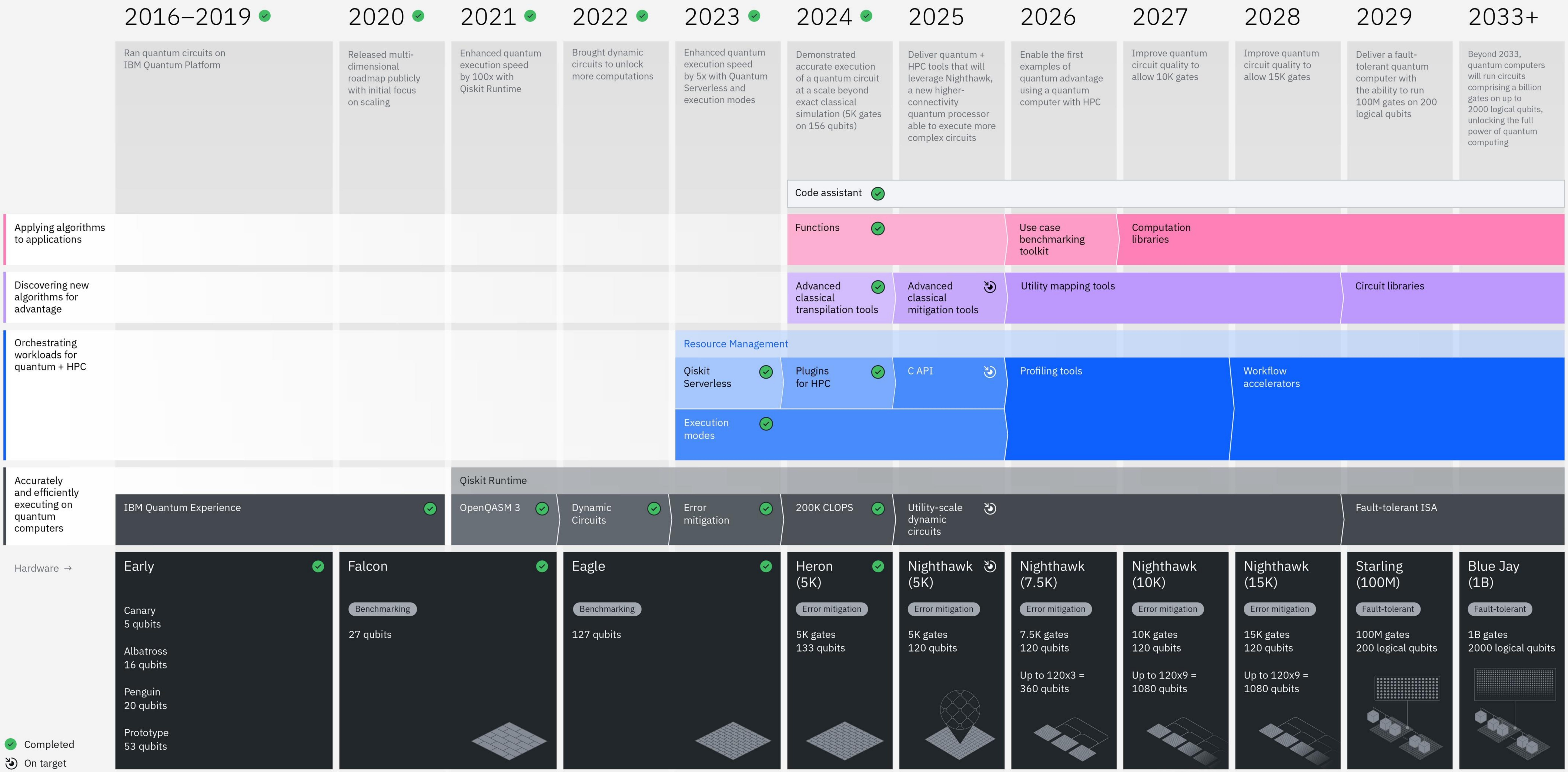
A scientific experiment is only as good as confidence in methods.

2
Quantum advantage is not something that will happen at a singular moment in time.

It is a hypothesis that is subject to falsification.

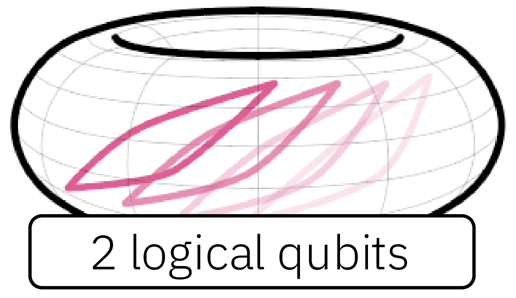
Development Roadmap

IBM **Quantum**



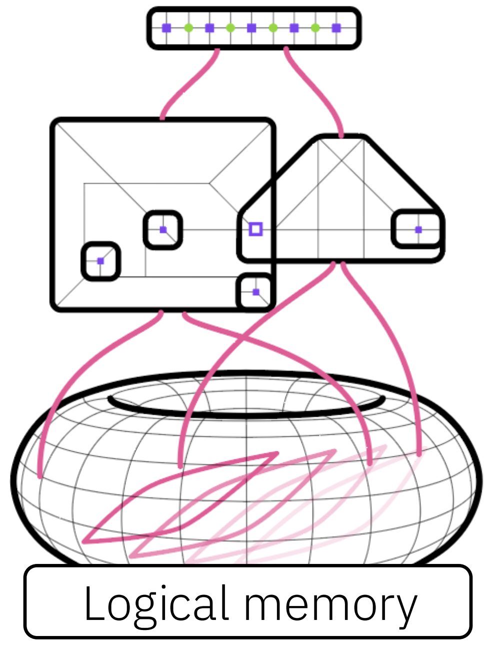
IBM fault-tolerant quantum computing roadmap

Loon (2025)



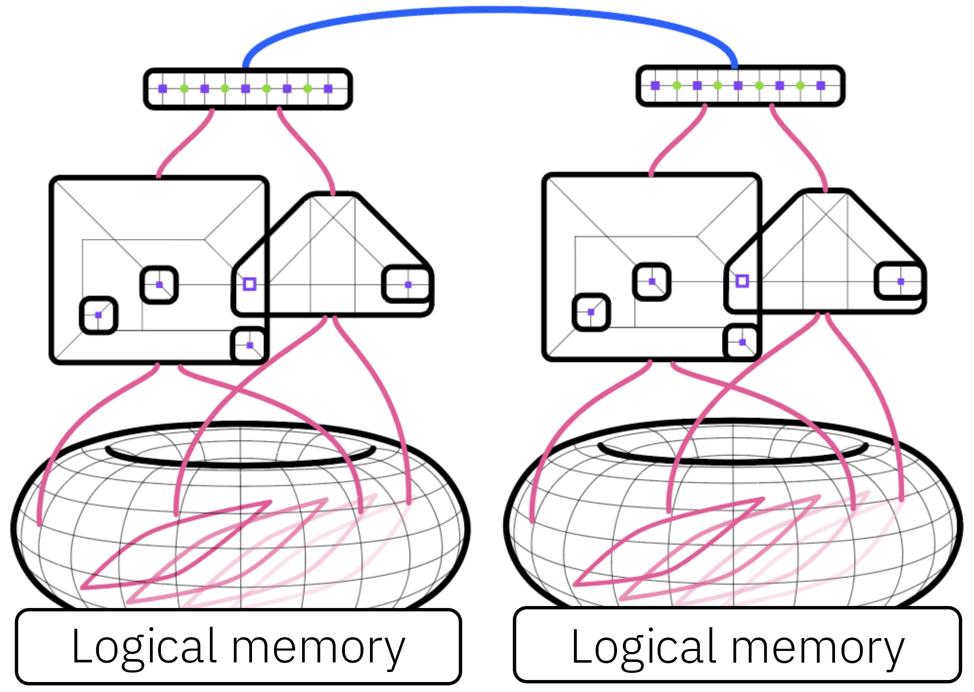
- 6-way connectivity
- c-coupler demo
- Automated design

Kookaburra (2026)



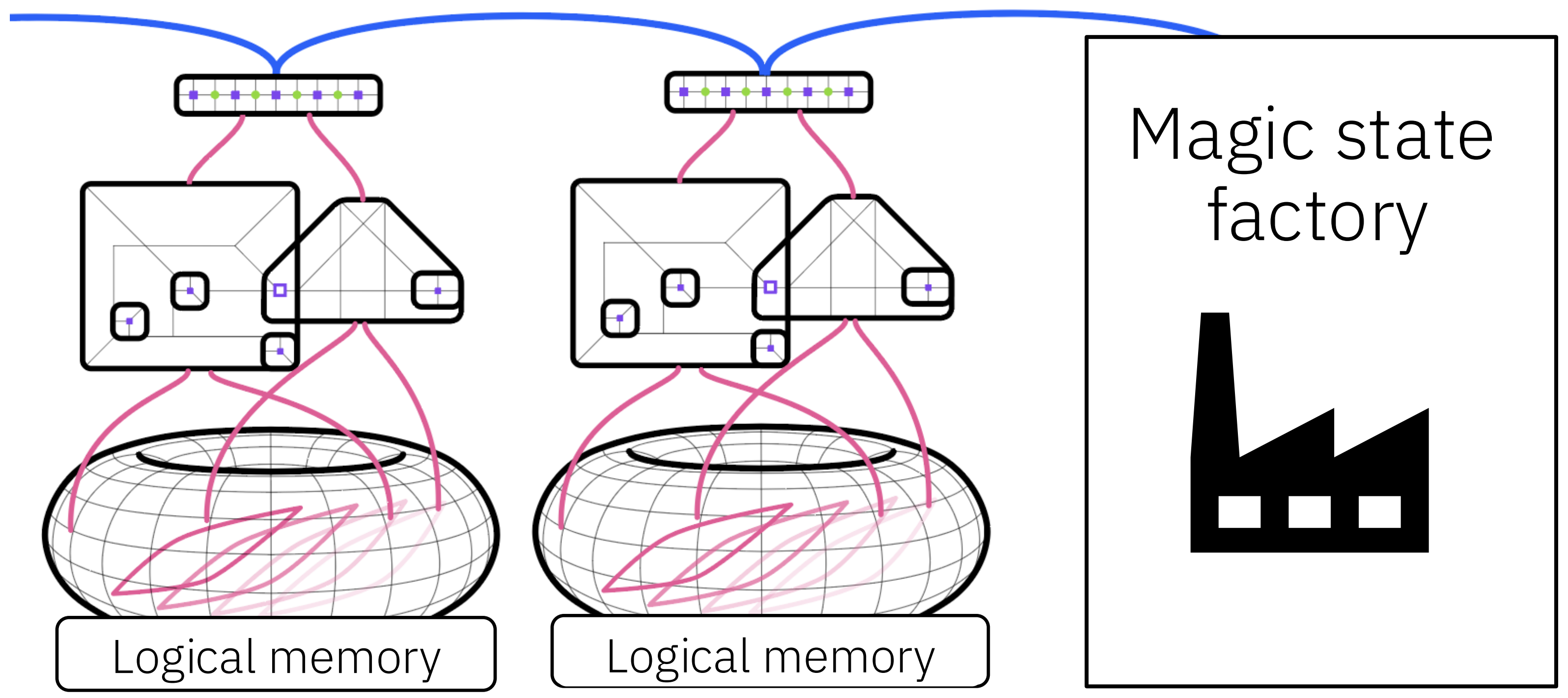
- “Long” c-couplers
- Real-time decoding
- LPU (+~100 qubits) for logical operations

Cockatoo (2027)



- Two blocks of gross code + LPU
- Module-to-module logical communication over l-couplers

Starling (2028)



- Multiple blocks of gross code + LPUs
- Universal computation with magic state distillation

Development Roadmap

IBM Quantum

2016–2019 ✓	2020 ✓	2021 ✓	2022 ✓	2023 ✓	2024 ✓	2025	2026	2027	2028	2029	2033+
Ran quantum circuits on IBM Quantum Platform	Released multi-dimensional roadmap publicly with initial focus on scaling	Enhanced quantum execution speed by 100x with Qiskit Runtime	Brought dynamic circuits to unlock more computations	Enhanced quantum execution speed by 5x with Quantum Serverless and execution modes	Demonstrated accurate execution of a quantum circuit at a scale beyond exact classical simulation (5K gates on 156 qubits)	Deliver quantum + HPC tools that will leverage Nighthawk, a new higher-connectivity quantum processor able to execute more complex circuits	Enable the first examples of quantum advantage using a quantum computer with HPC	Improve quantum circuit quality to allow 10K gates	Improve quantum circuit quality to allow 15K gates	Deliver a fault-tolerant quantum computer with the ability to run 100M gates on 200 logical qubits	Beyond 2033, quantum computers will run circuits comprising a billion gates on up to 2000 logical qubits, unlocking the full power of quantum computing

Code assistant ✓ (2024 onward)

Applying algorithms to applications
- Functions ✓ (2024–2025)
- Use case benchmarking toolkit (2026)
- Computation libraries (2027 onward)

Discovering new algorithms for advantage
- Advanced classical transpilation tools ✓ (2024)
- Advanced classical mitigation tools (on target) (2025)
- Utility mapping tools (2026–2028)
- Circuit libraries (2029 onward)

Orchestrating workloads for quantum + HPC
- Resource Management (2023 onward)
- Qiskit Serverless ✓ (2023)
- Plugins for HPC ✓ (2024)
- C API (on target) (2025)
- Execution modes ✓ (2023–2025)
- Profiling tools (2026–2027)
- Workflow accelerators (2028 onward)

Accurately and efficiently executing on quantum computers
- Qiskit Runtime (2021 onward)
- IBM Quantum Experience ✓ (2016–2020)
- OpenQASM 3 ✓ (2021)
- Dynamic Circuits ✓ (2022)
- Error mitigation ✓ (2023)
- 200K CLOPS ✓ (2024)
- Utility-scale dynamic circuits (on target) (2025–2028)
- Fault-tolerant ISA (2029 onward)

Hardware →

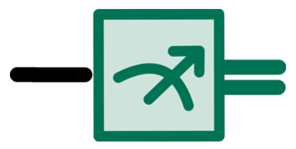
2016–2019	Early ✓		Canary 5 qubits; Albatross 16 qubits; Penguin 20 qubits; Prototype 53 qubits
2020	Falcon ✓	Benchmarking	27 qubits
2022	Eagle ✓	Benchmarking	127 qubits
2024	Heron (5K) ✓	Error mitigation	5K gates, 133 qubits
2025	Nighthawk (5K) (on target)	Error mitigation	5K gates, 120 qubits
2026	Nighthawk (7.5K)	Error mitigation	7.5K gates, 120 qubits; Up to 120x3 = 360 qubits
2027	Nighthawk (10K)	Error mitigation	10K gates, 120 qubits; Up to 120x9 = 1080 qubits
2028	Nighthawk (15K)	Error mitigation	15K gates, 120 qubits; Up to 120x9 = 1080 qubits
2029	Starling (100M)	Fault-tolerant	100M gates, 200 logical qubits
2033+	Blue Jay (1B)	Fault-tolerant	1B gates, 2000 logical qubits

✓ Completed
On target

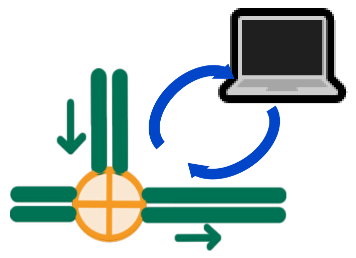
What are Dynamic Circuits?

Dynamic circuits are quantum circuits that incorporate

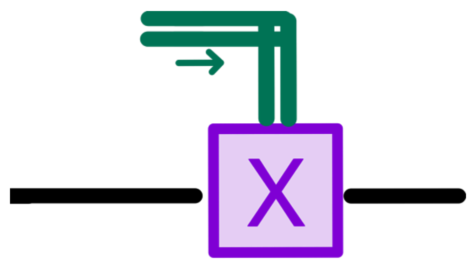
Mid-circuit measurements

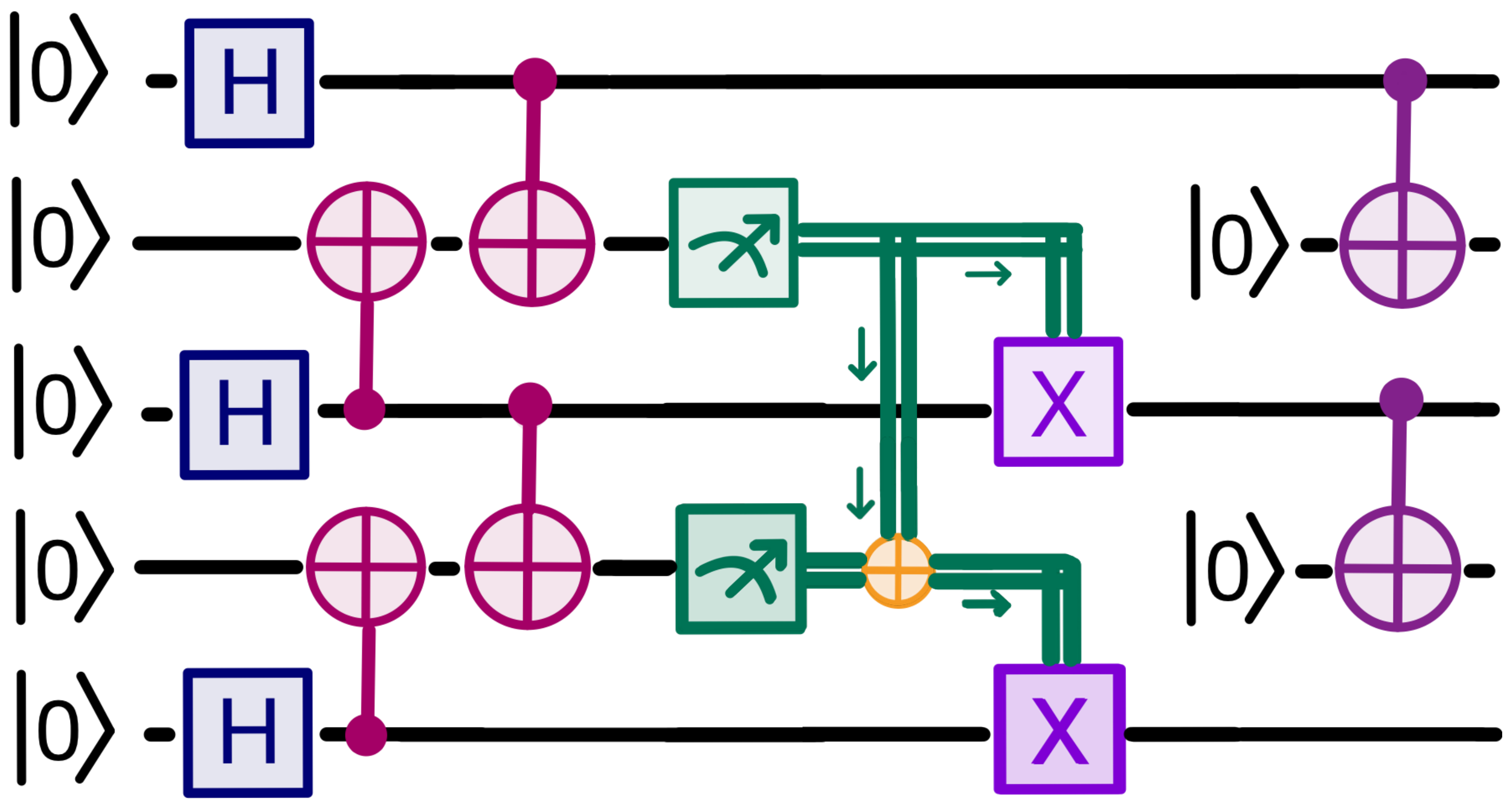


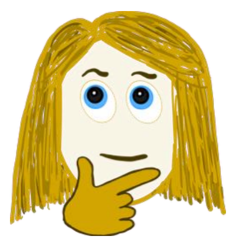
Classical calculations based on the measurement results



Feed-forward operations





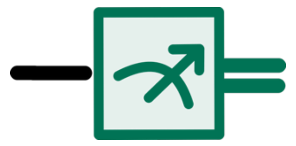


But why is that beneficial?

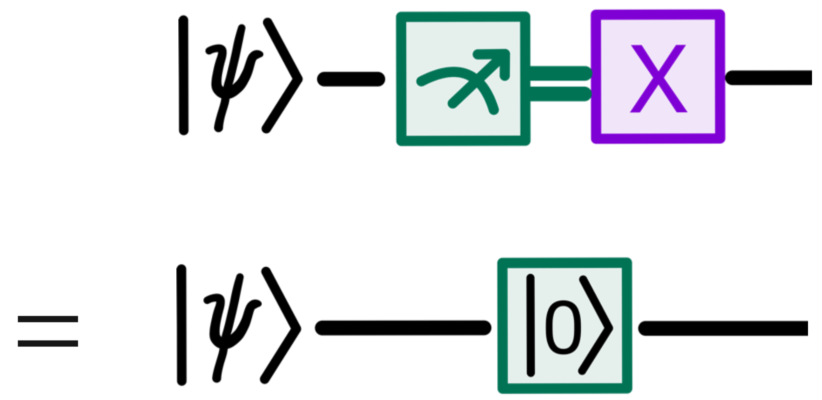
Why Dynamic Circuits?

Dynamic circuits are quantum circuits that incorporate

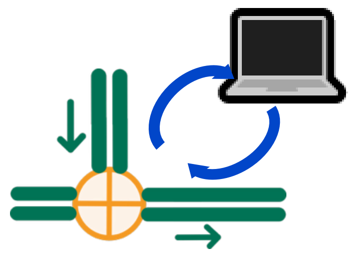
Mid-circuit measurements



- Conditional reset:

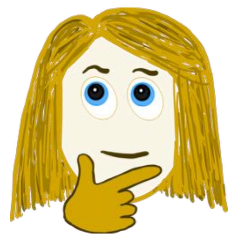


Classical calculations based on the measurement results



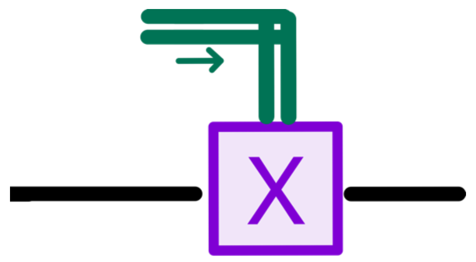
- For certain problems, quantum computers are exptected to give an advantage
- But: in general, classical computations are much faster, have much higher fidelity and no connectivity constraints

➡ Hybrid classical-quantum algorithms like VQE etc



But measurements make the state collapse?

Feed-forward operations



- Famous application: Error Correction

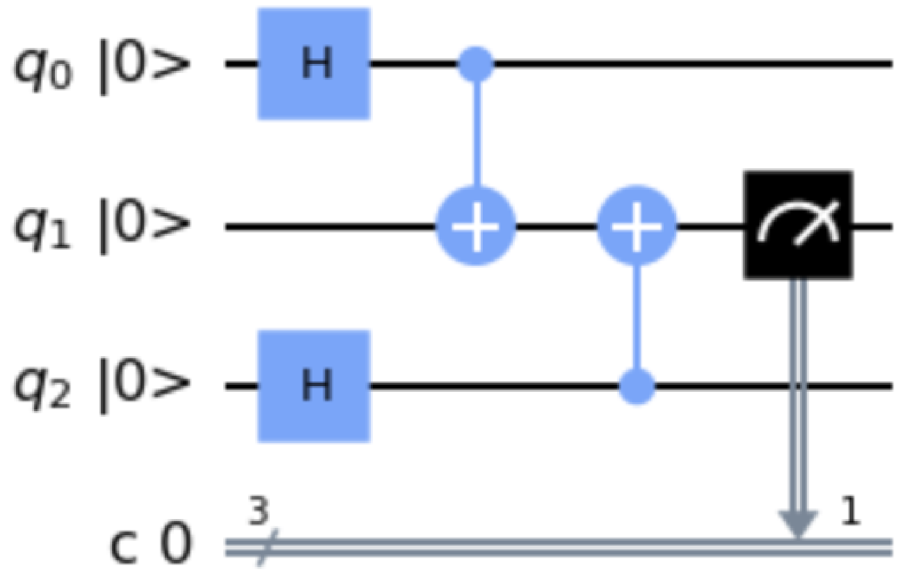
➡ Check for errors and directly correct them

- Use classical calculations as information transfer to spread correlations faster

➡ Create long-range entanglement in a **shallow quantum circuit**

Dynamic Circuits for Long-Range Entanglement

- Let us consider the following example:

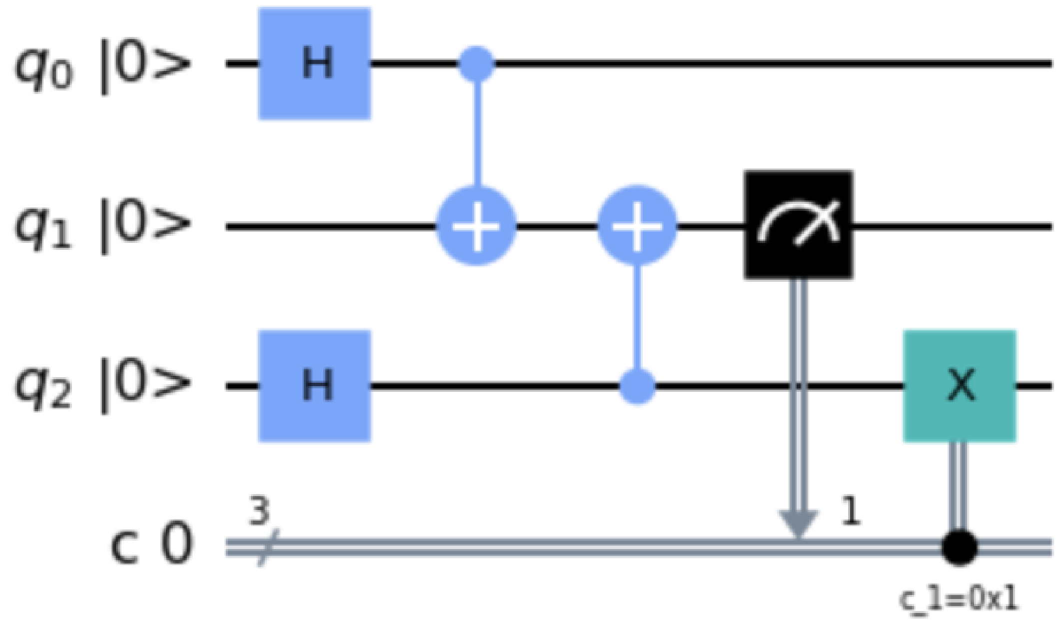


- post-measurement state either

$$\frac{1}{\sqrt{2}}(|00\rangle_{AC} + |11\rangle_{AC}) \text{ or } \frac{1}{\sqrt{2}}(|01\rangle_{AC} + |10\rangle_{AC})$$

  - ➢ actually a mixed state

- If we apply a conditional gate:



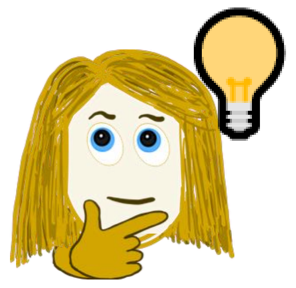
- Final state

$$\frac{1}{\sqrt{2}}(|00\rangle_{AC} + |11\rangle_{AC})$$

→ using dynamic circuits we can create entanglement without a direct link in constant depth

→ especially useful for quantum devices with limited connectivity!

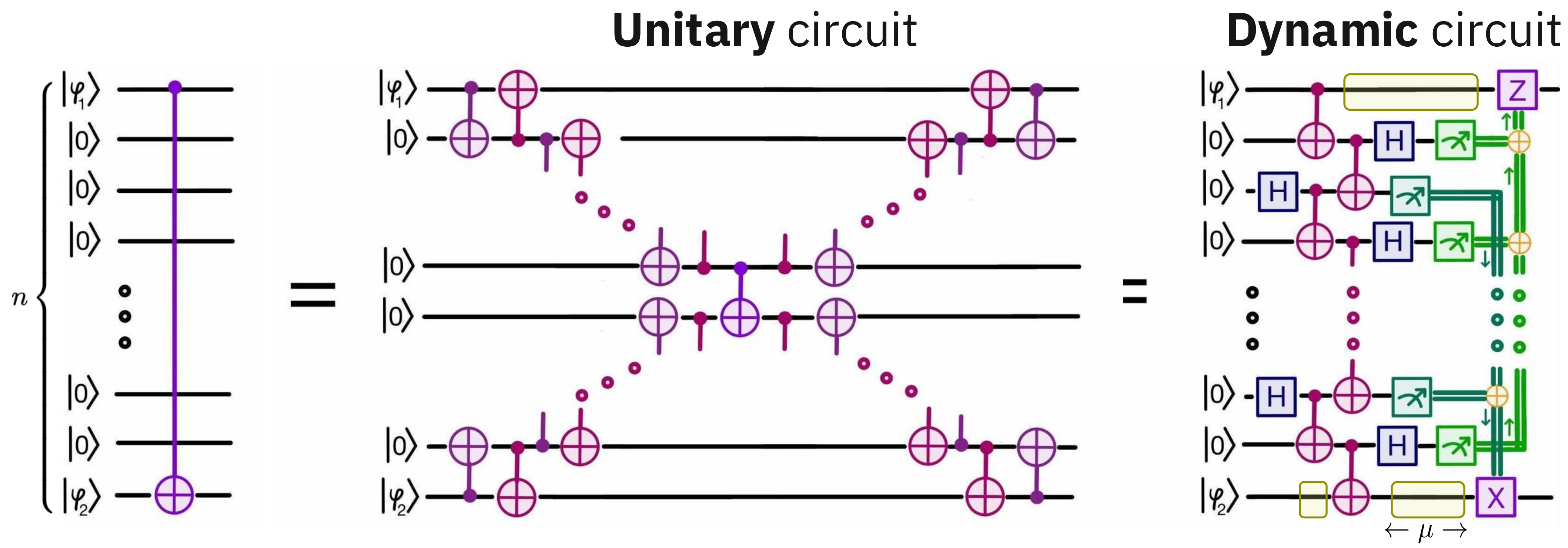
But why real-time feed-forward and not just post-processing?



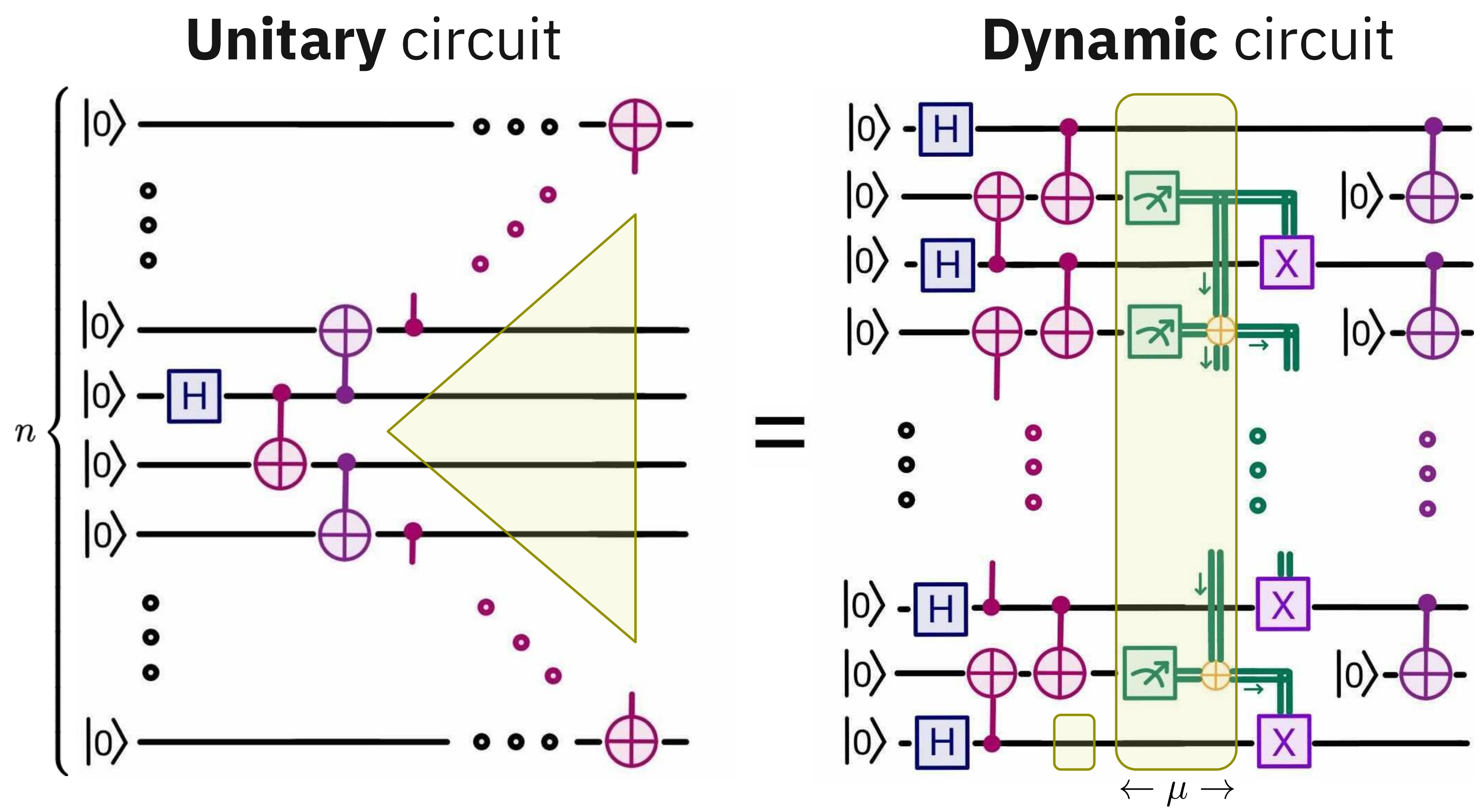
Post-processing only works if we can simulate the different outcomes through the whole circuit!

Dynamic Circuits for Long-Range Entanglement

- Implementation of long-range entangling gates due to local connectivity
  - to entangle two qubits that are distance n away we need depth $\mathcal{O}(n)$



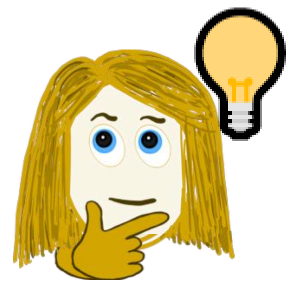
- Preparation of long-ranged entangled states
  - to entangle n qubits, we need at least depth $\mathcal{O}(\log n)$



EB, V. Tripathi, D. Wang, P. Rall, E. Chen, S. Majumder, A. Seif, Z. Minev, https://arxiv.org/abs/2308.13065

- Using dynamic circuits, we can create long-range entanglement in a **constant depth quantum circuit**

But entanglement cannot spread faster than the information light-cone?



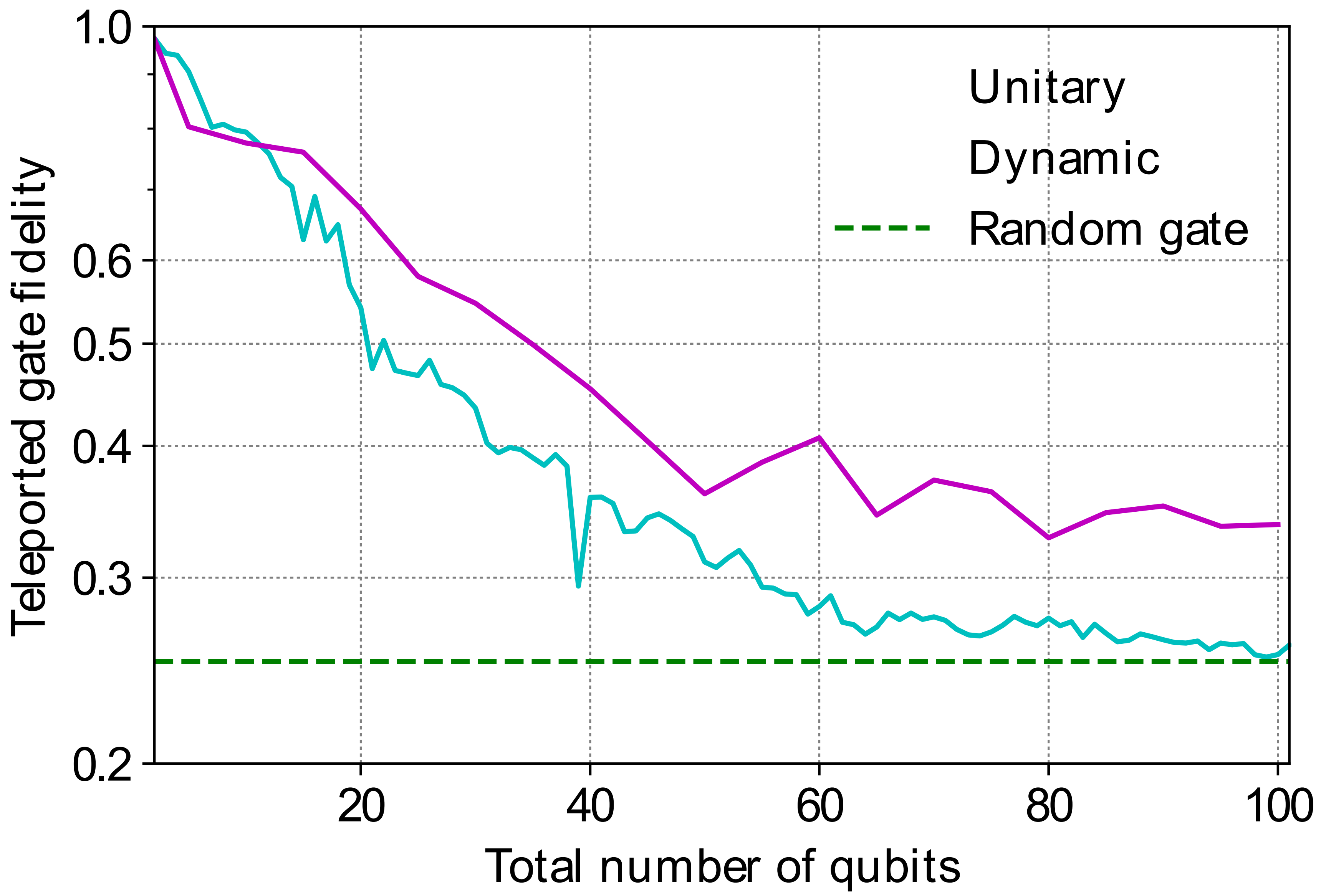
True, but the information transfer in this case is performed by the classical calculation!

Experimental Results

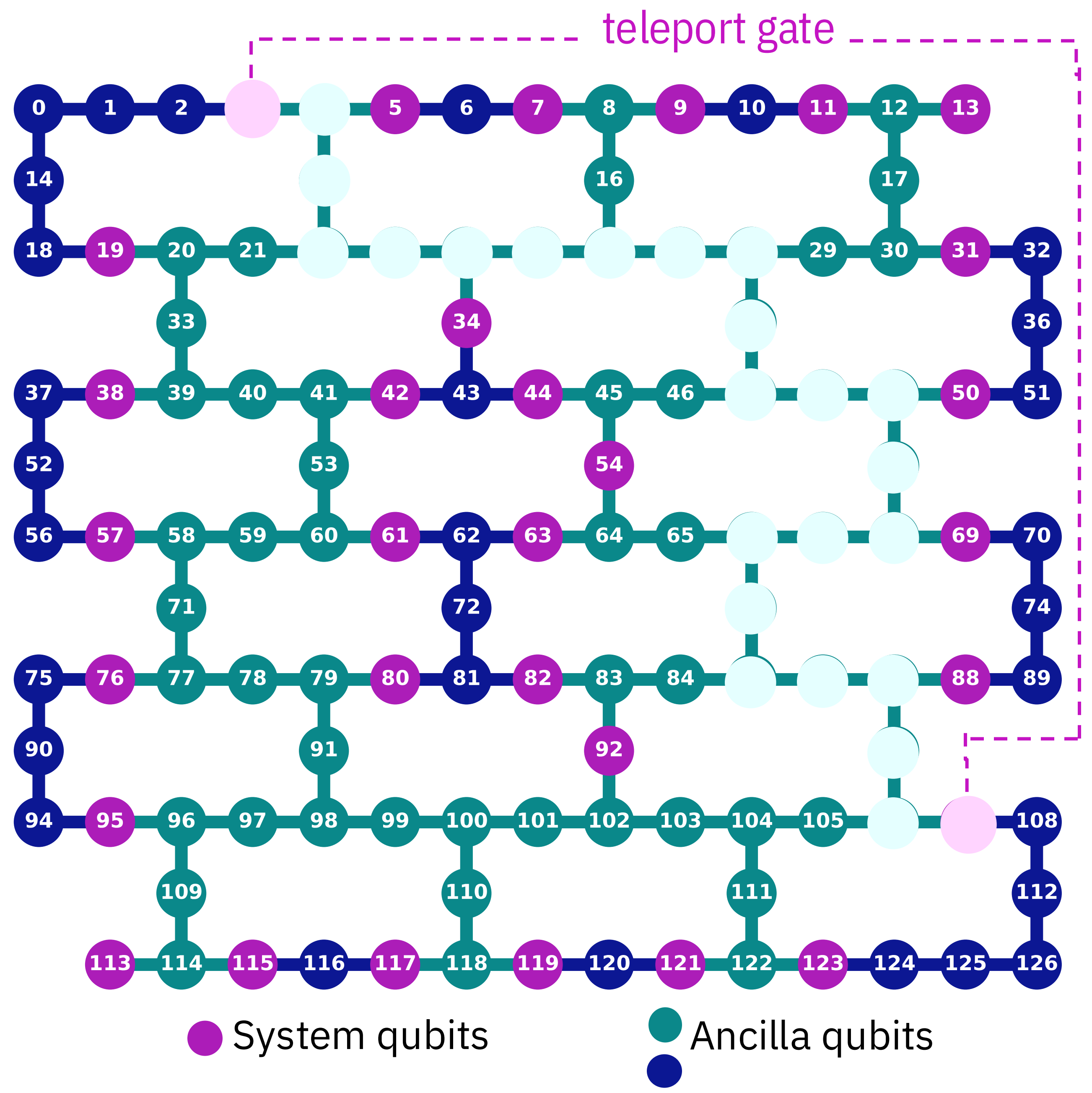
PRX Quantum **5**, 030339

Joint work with Vinay Tripathi, Derek Wang, Patrick Rall, Edward Chen, Swarnadeep Majumder, Alireza Seif & Zlatko Minev

- CNOT gates over large distances are more efficiently executed with dynamic circuits than unitary ones



- Outlook: Teleporting gates can provide a workaround for effective all-to-all connectivity



Dynamic circuits are a promising feature to help overcome current limitations of hardware, especially for increasing number of qubits

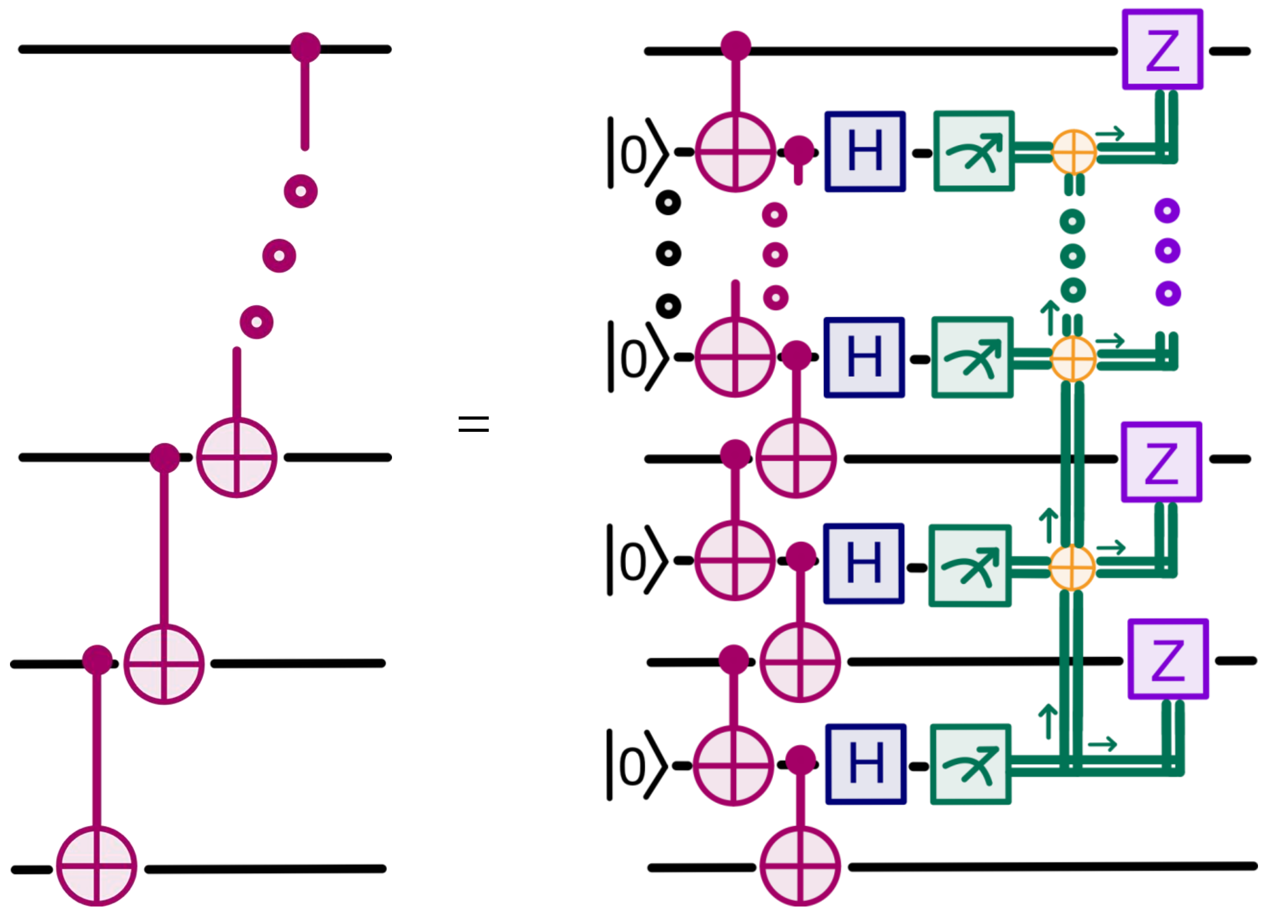
Measurement-Based Long-Range Entangling Gates in Constant Depth

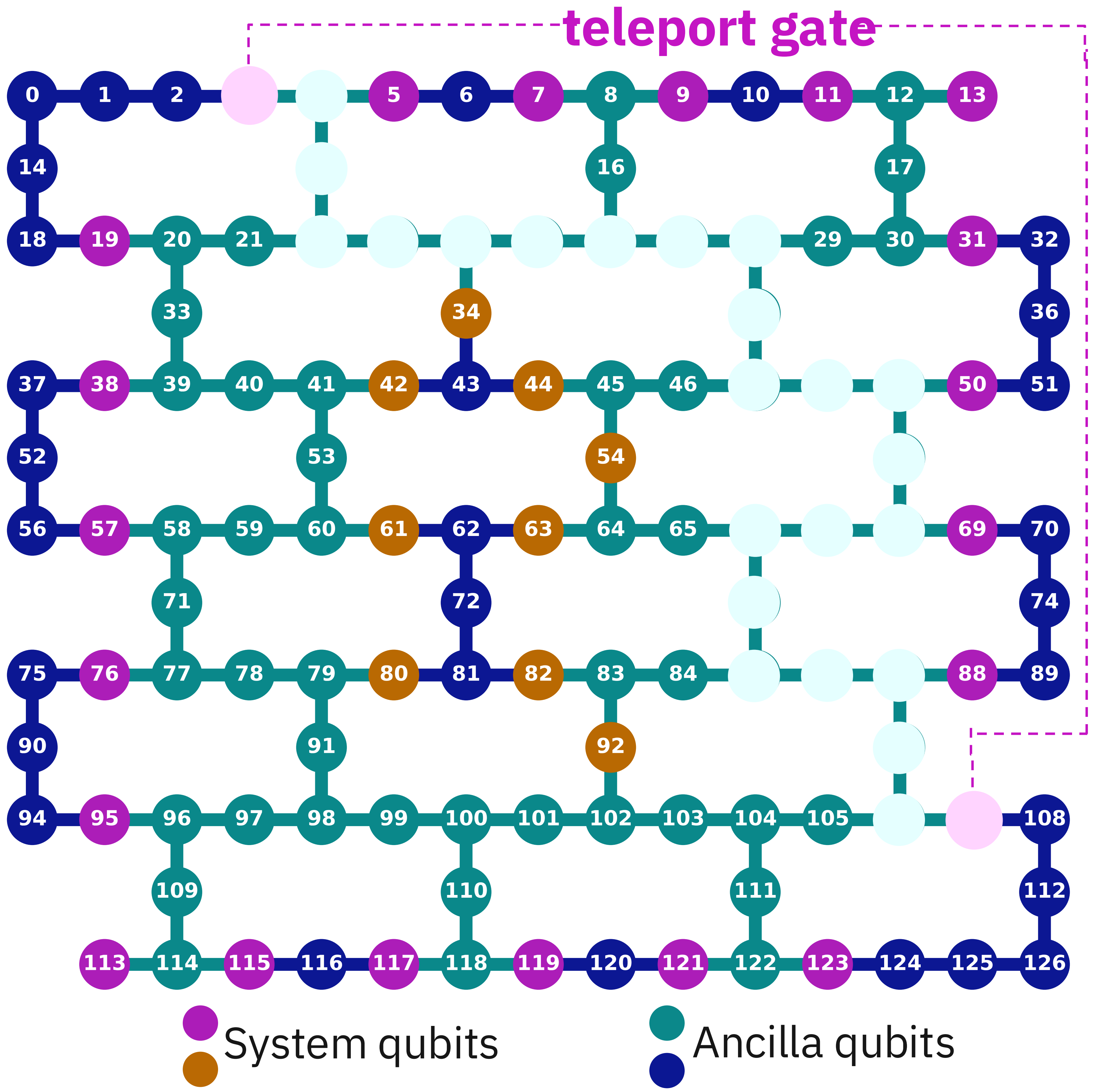
arXiv:2408.03064

Joint work with Stefan Wörner

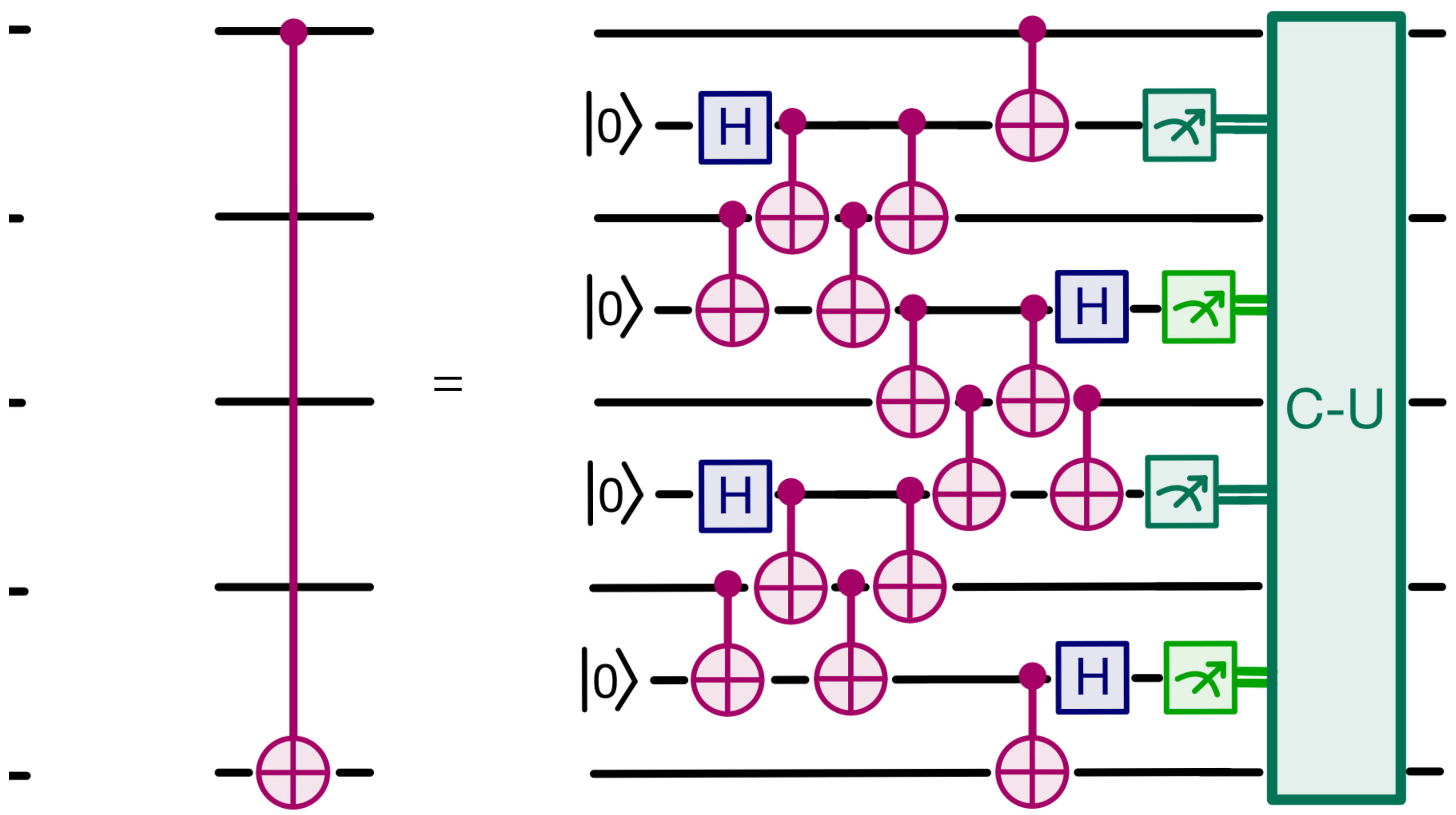
- Setting: $n+1$ system qubits
- ➢ Constructions for quantum s … ınt depth using dynamic circuits:

"CNOT ladder":





Long-range CNOT gate:



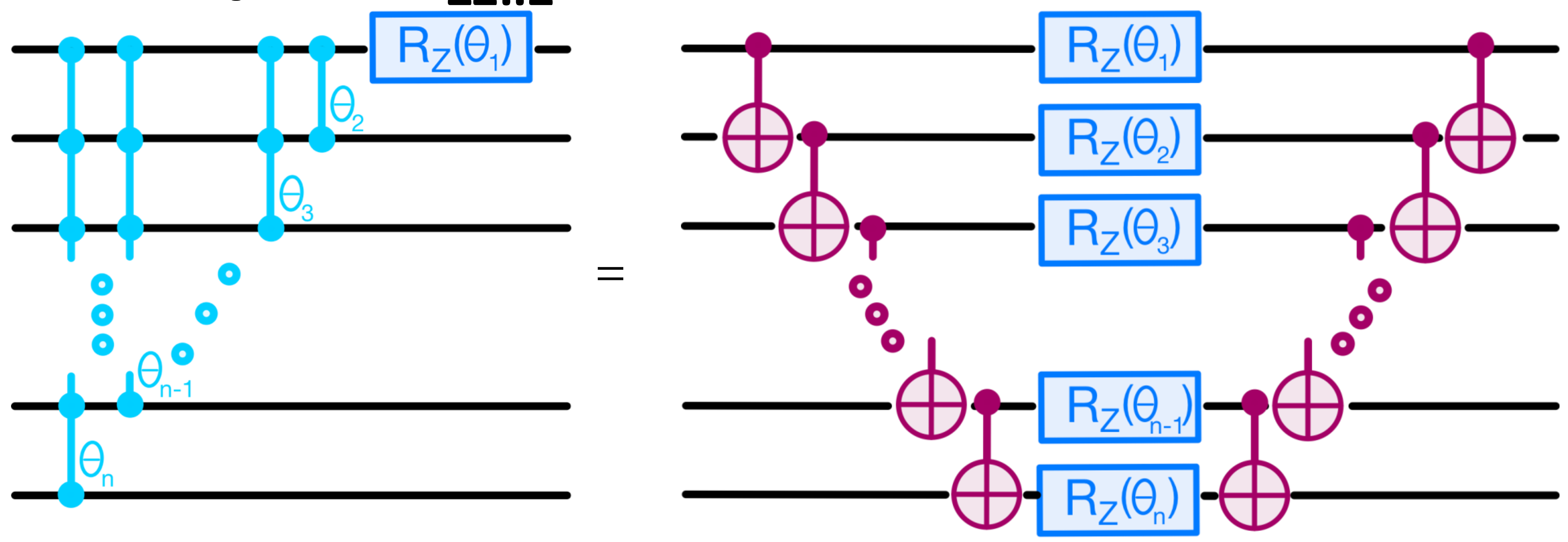
Measurement-Based Long-Range Entangling Gates in Constant Depth

arXiv:2408.03064

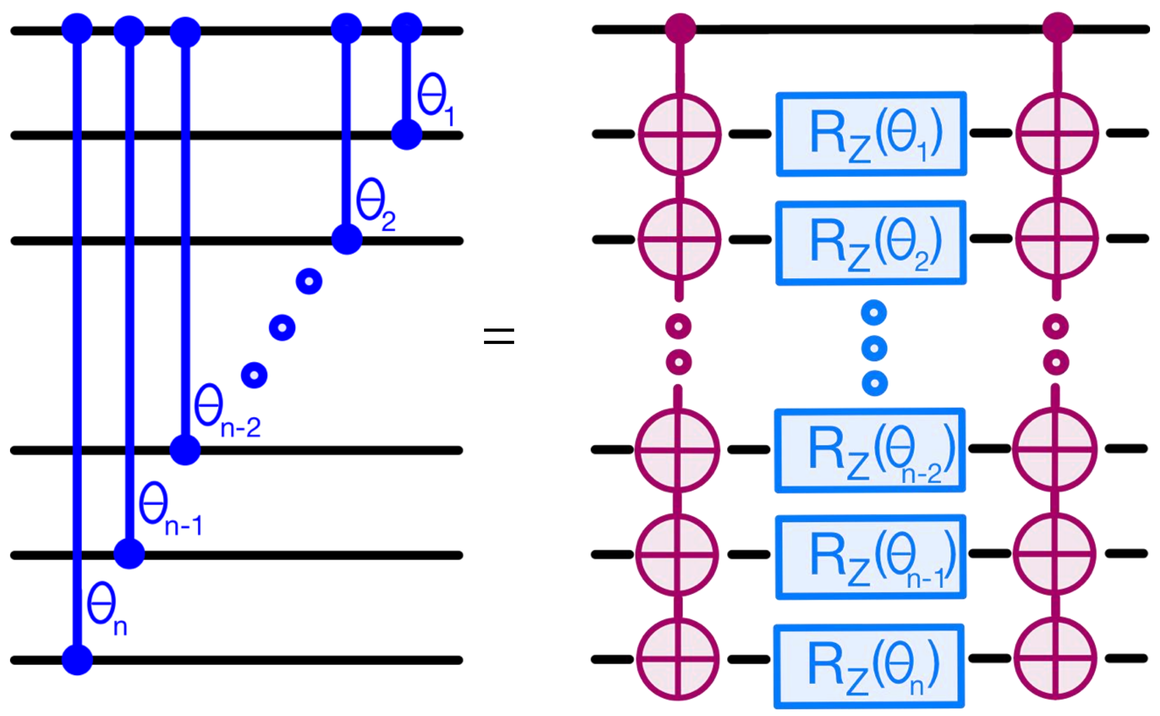
Joint work with Stefan Wörner

- These constructions can be combined to construct multi-qubit (parametrized) rotation gates:

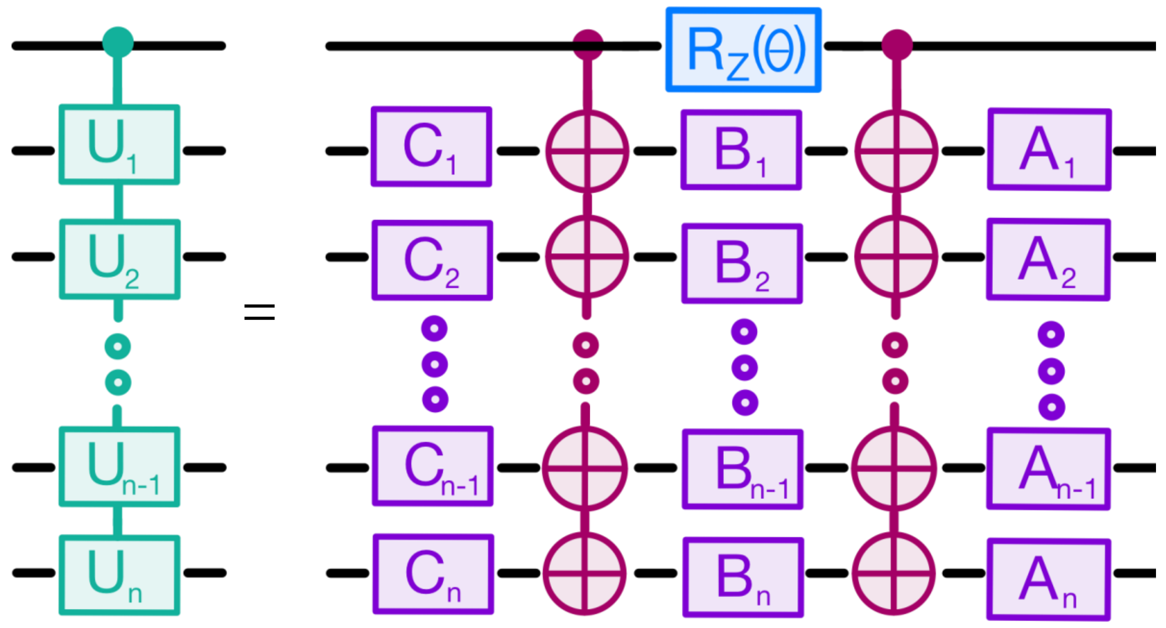
Multi-qubit $R_{ZZ..Z}$ rotations:



Parallel R_{ZZ} rotations:



Generalized fan-out gate:



Experimental Results

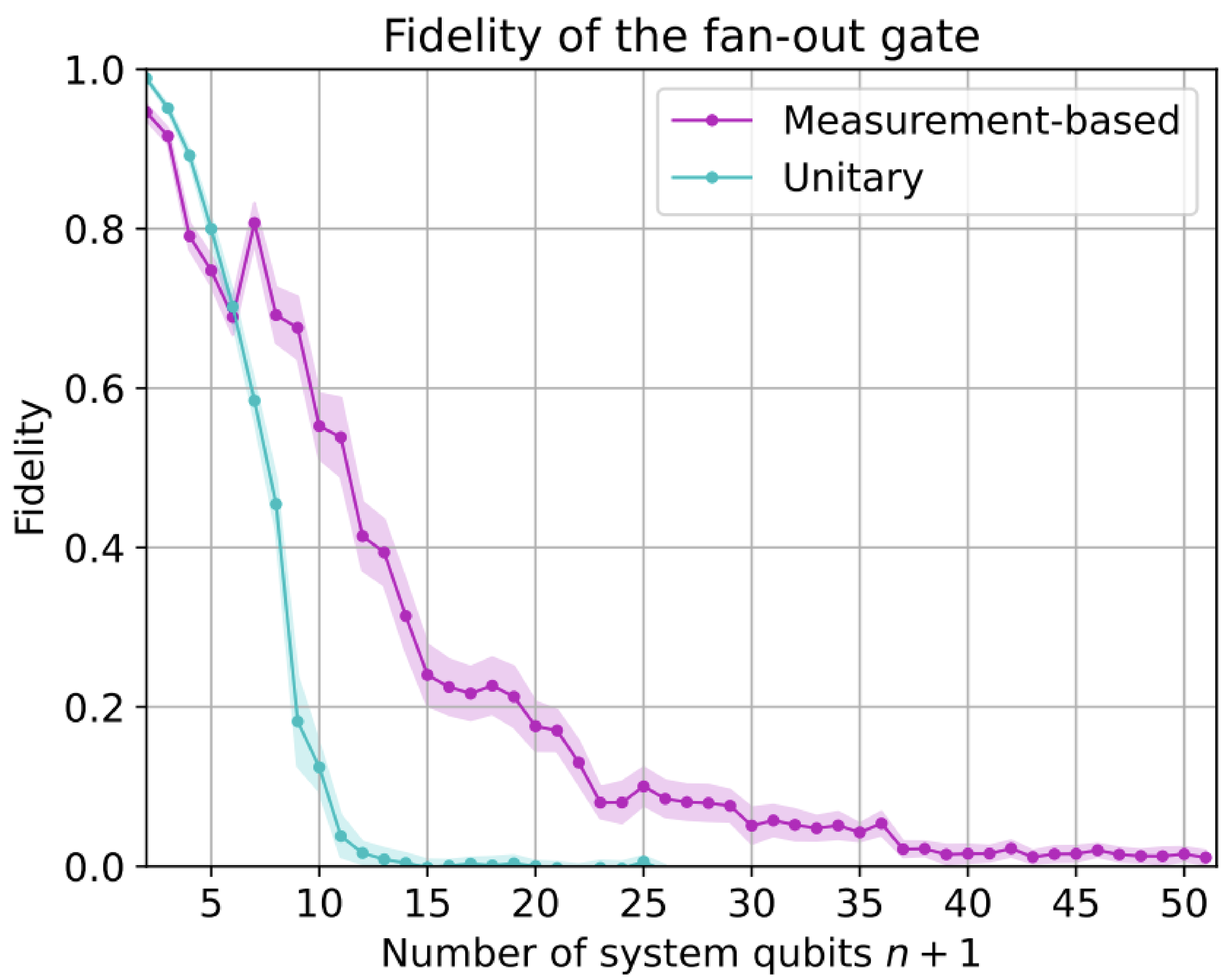


FIG. 5. Experimental results on `ibm_kyiv` [13] implementing the fan-out gate on $n+1$ qubits ($2n+1$ total qubits in the measurement-based implementation).

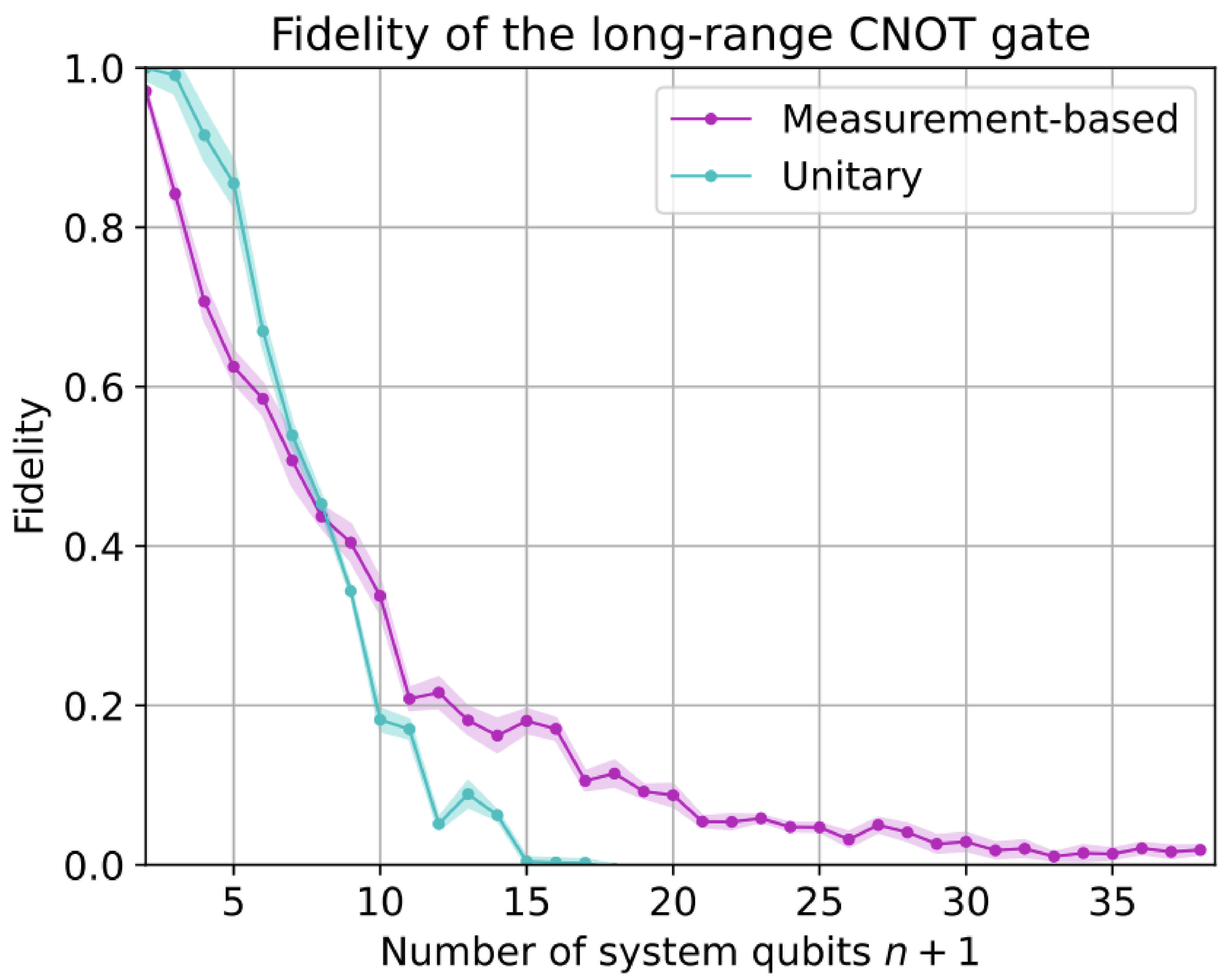


FIG. 6. Experimental results on `ibm_kyiv` [13] implementing the long-range CNOT gate on $n + 1$ qubits ($2n + 1$ total qubits in the measurement-based implementation).

➢ The measurement-based protocol outperforms the unitary one for larger number of qubits

Phys. Rev. Lett. **133**, 150602

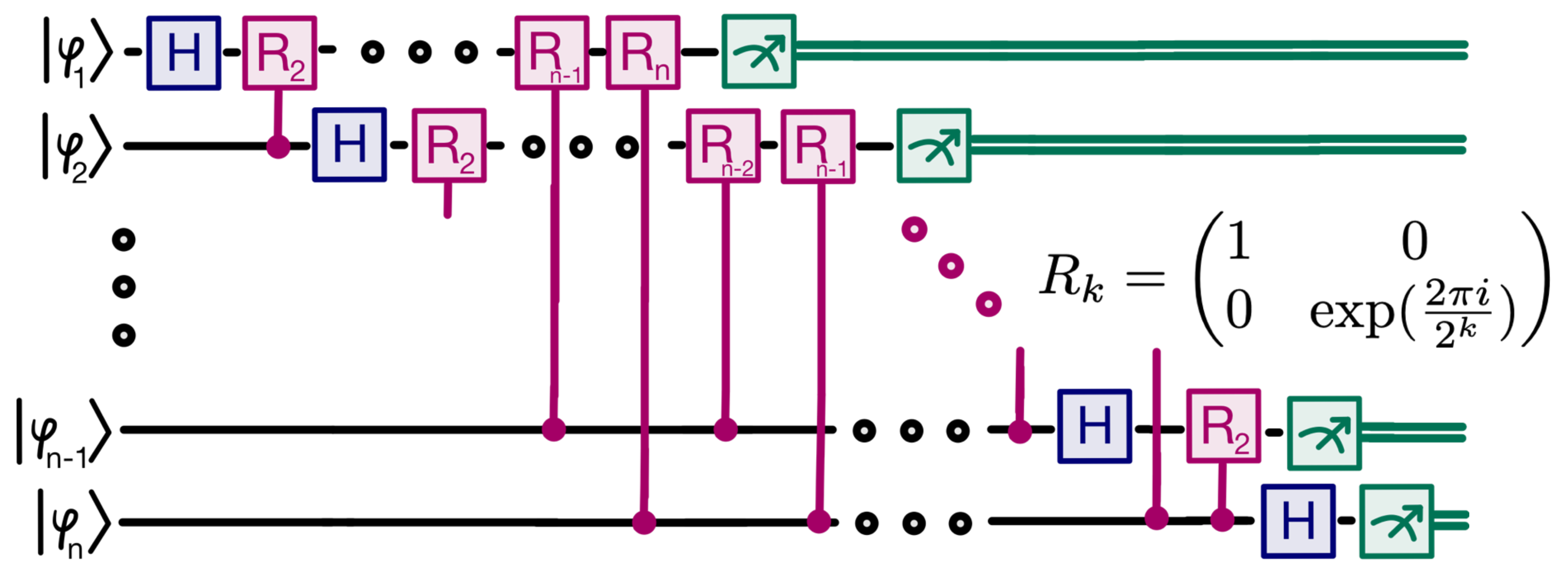
Quantum Fourier Transform using Dynamic Circuits

Joint work with Vinay Tripathi, Alireza Seif, Daniel Lidar & Derek Wang

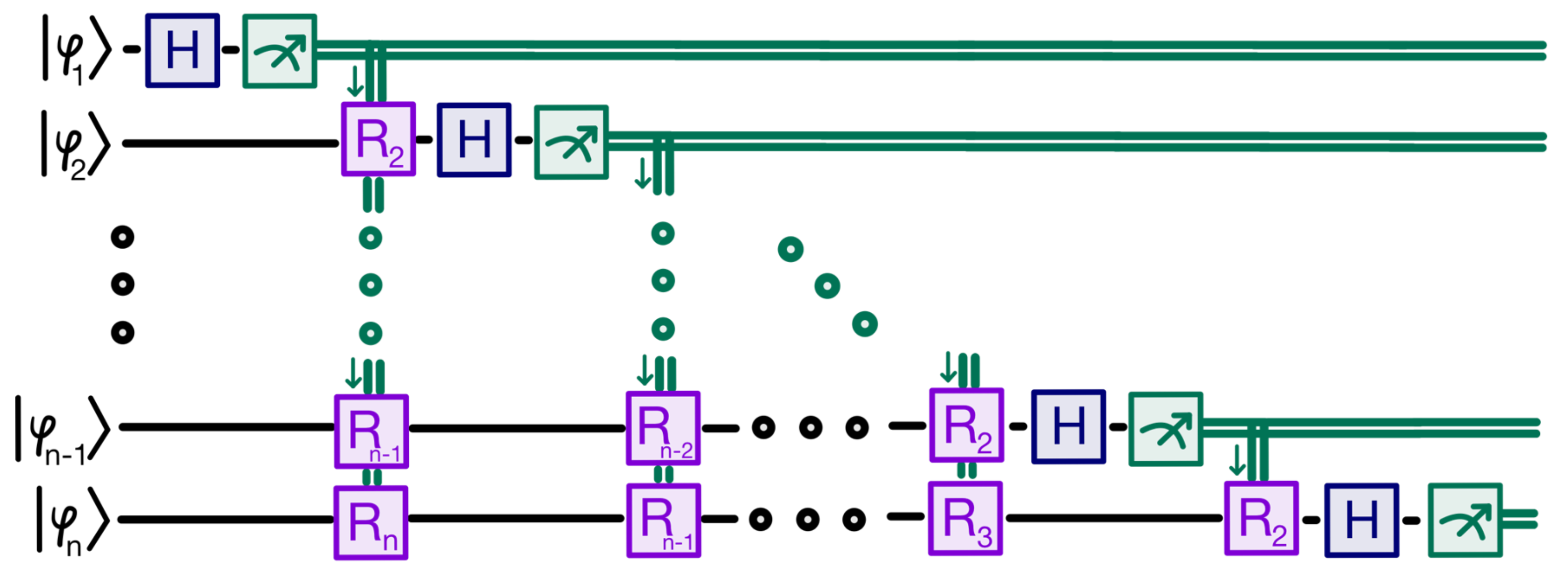
- Important subroutine of many quantum algorithms, e.g. Shor's Algorithm or Quantum Phase Estimation. It is often followed by a measurement and can then be drastically simplified using dynamic circuits.

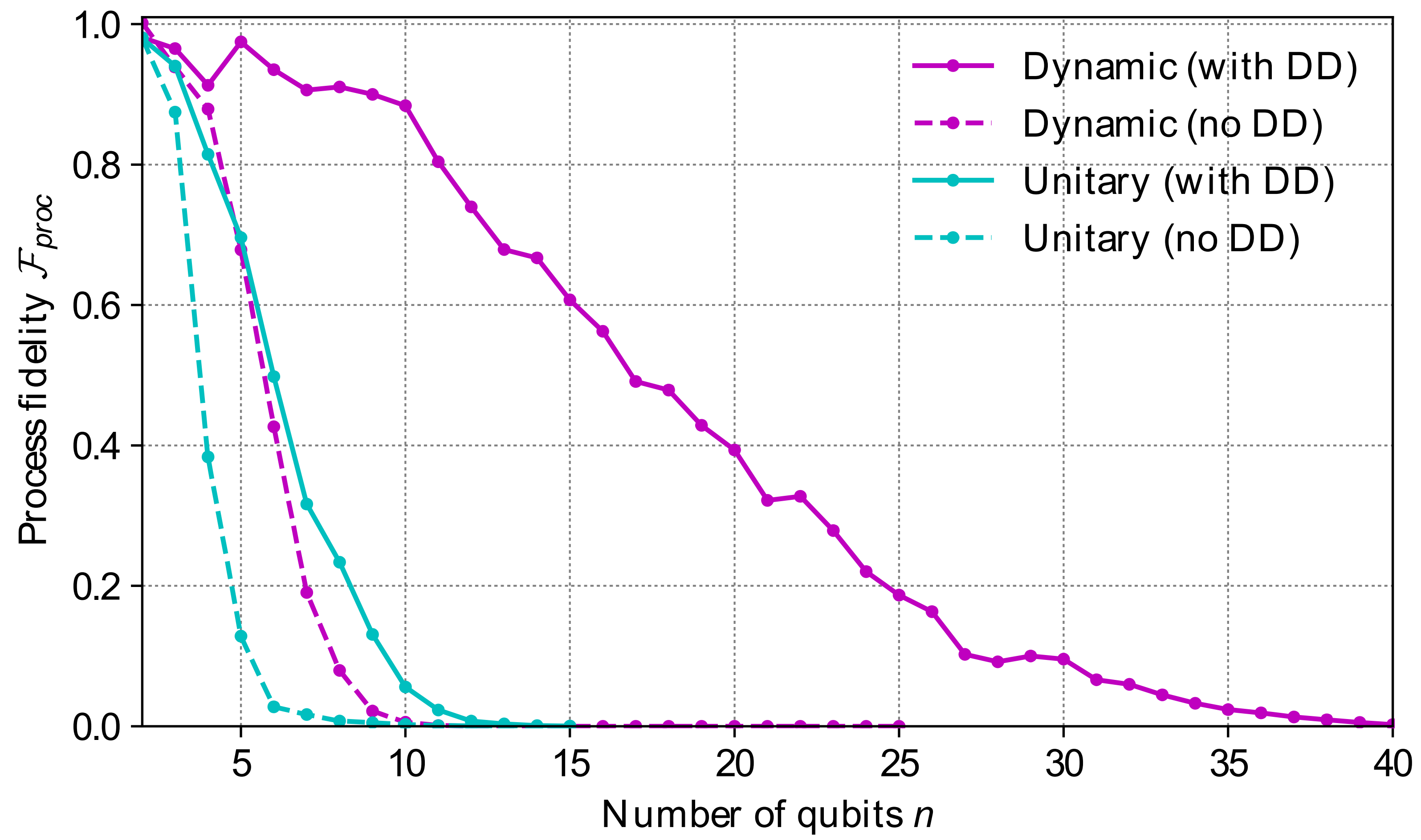
➢ Instead of $\mathcal{O}(n^2)$ two-qubit gates in the standard unitary circuit, we only require $\mathcal{O}(n)$ mid-circuit measurements in the dynamic circuit without any connectivity constraints.

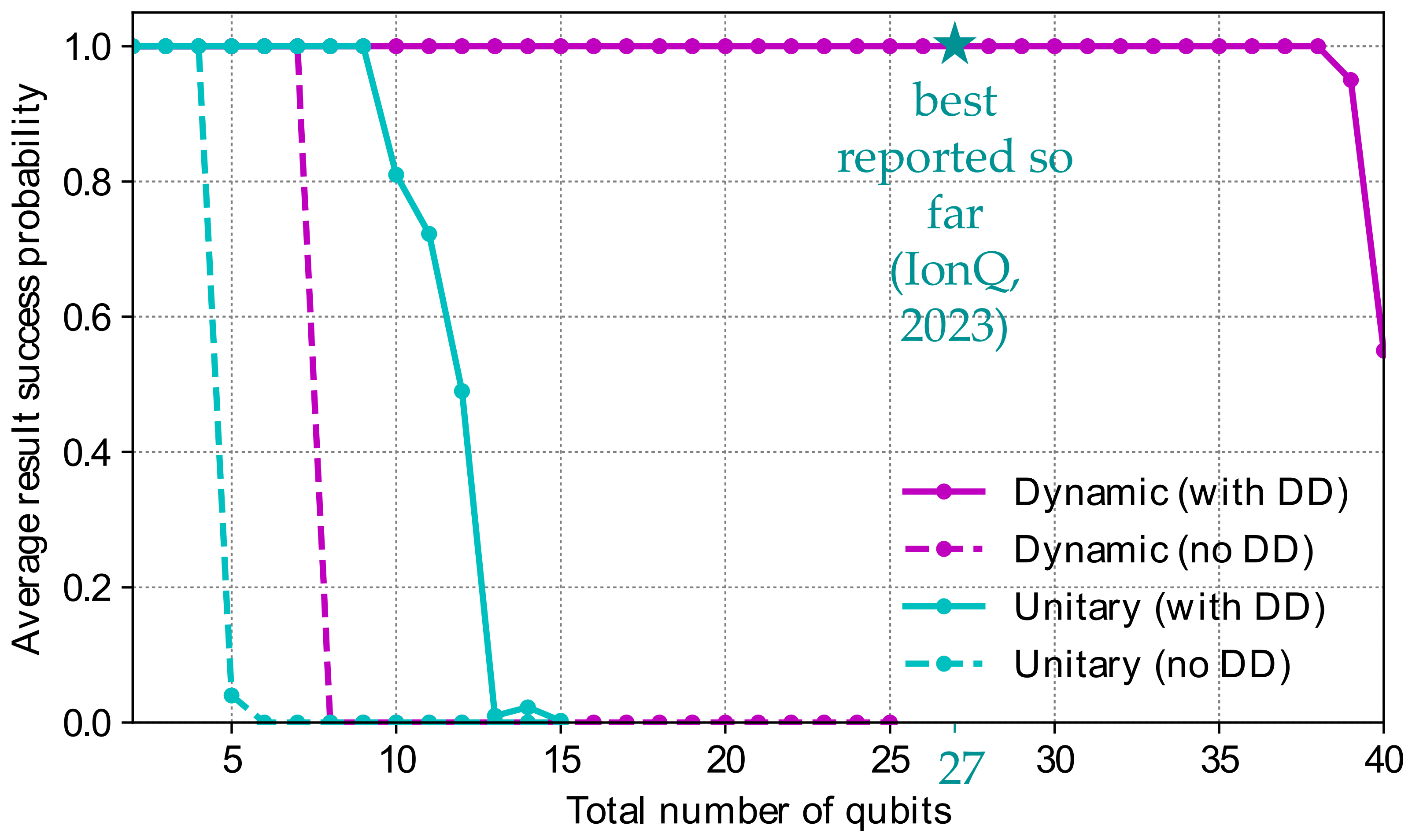
Unitary circuit



Dynamic circuit

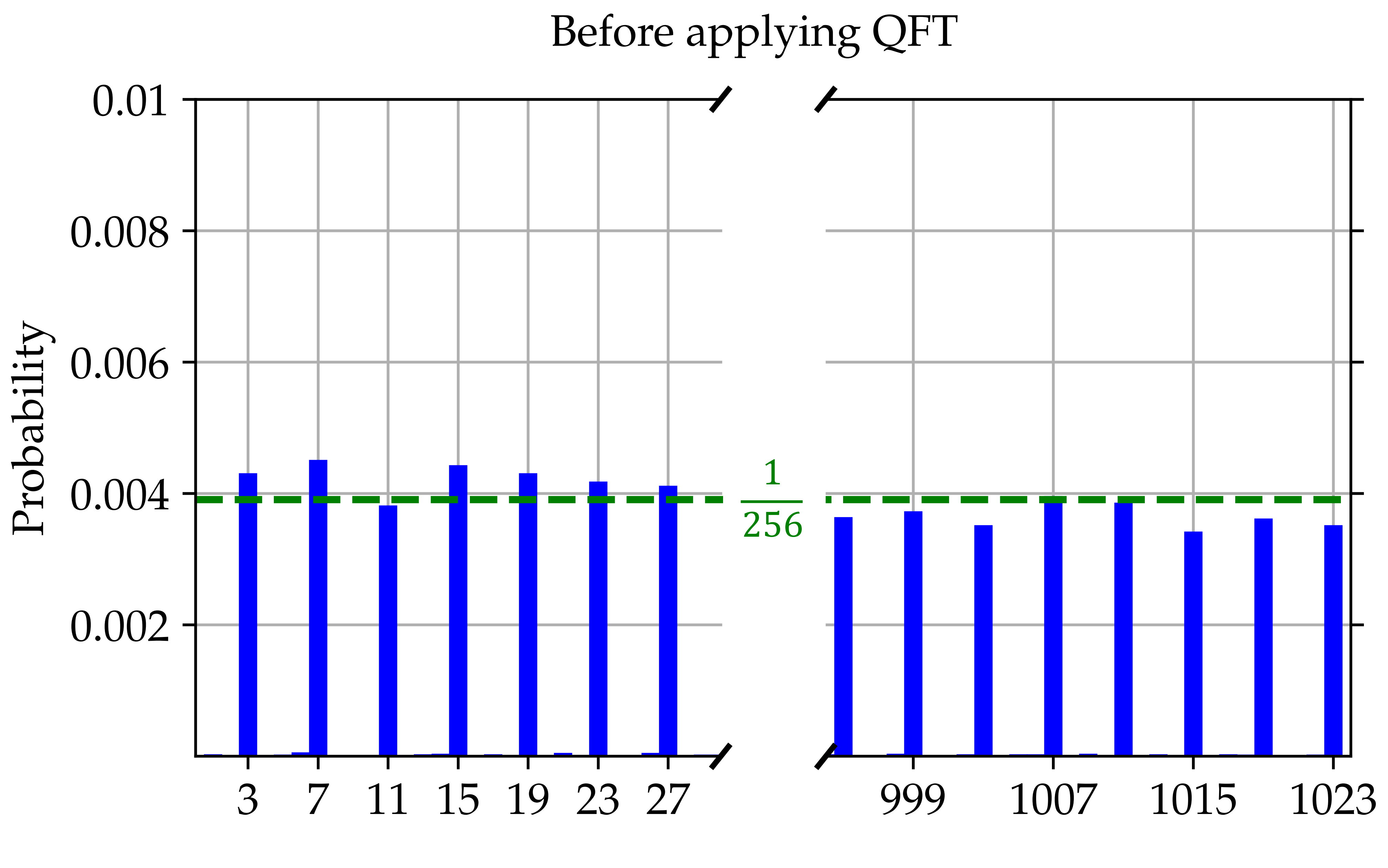


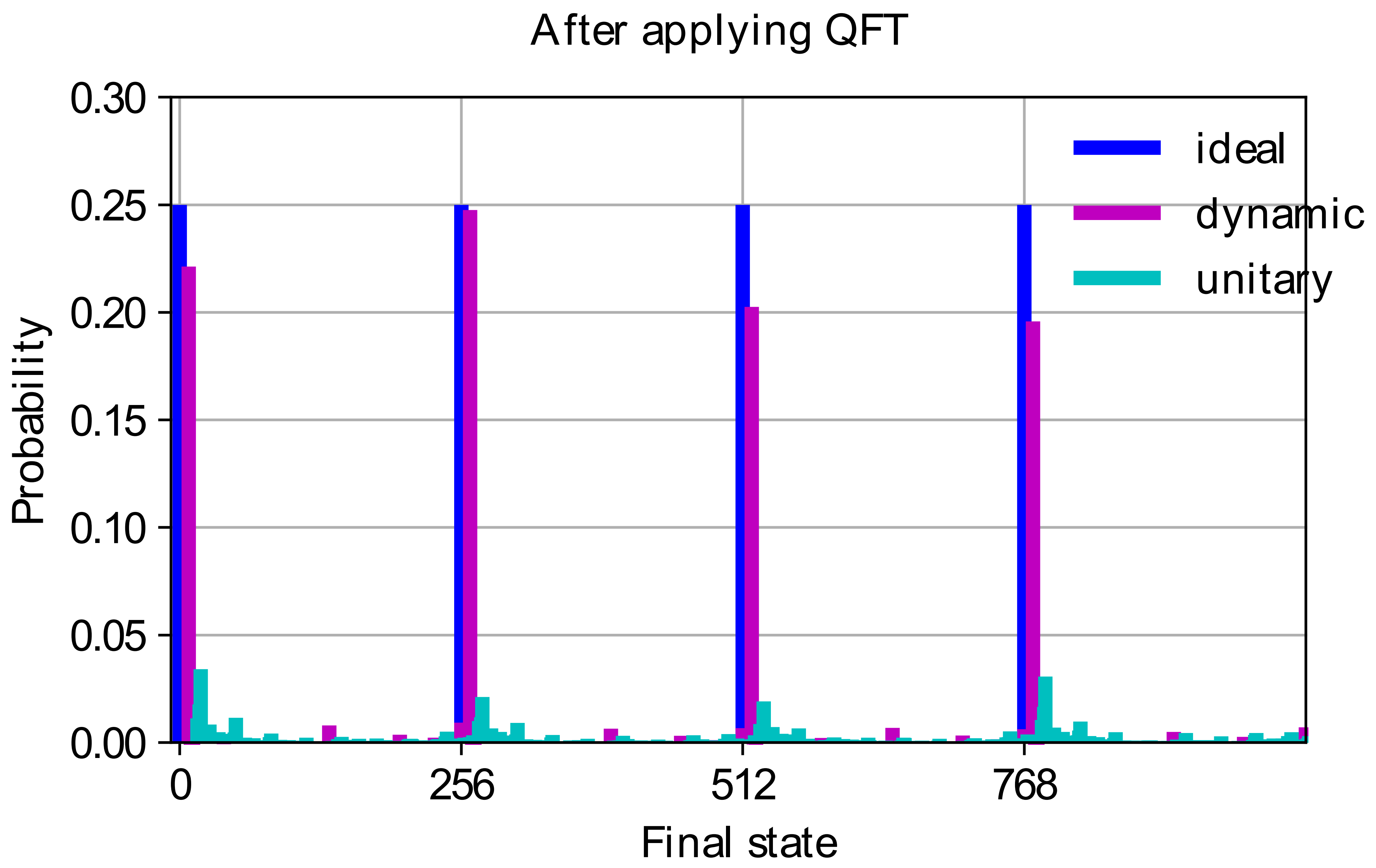




Experimental Results

- To visualize the improvement of QFT+M with dynamic circuits, we prepare a simple periodic state on 10 qubits and compare the ideal, unitary and dynamic circuit implementation of QFT+M applied to that state





➢ Implementation with dynamic circuits resembles ideal case, while unitary circuits result in a flatter distribution

arXiv:2504.20832

Approximate Quantum Fourier Transform in Logarithmic Depth on a Line

Joint work with David Sutter & Stefan Wörner

Approximate the Quantum Fourier Transform with error ε in depth $\mathcal{O}\left(\log \frac{n}{\varepsilon^2}\right)$ using only $\mathcal{O}(n)$ qubits on a line!

Construction:

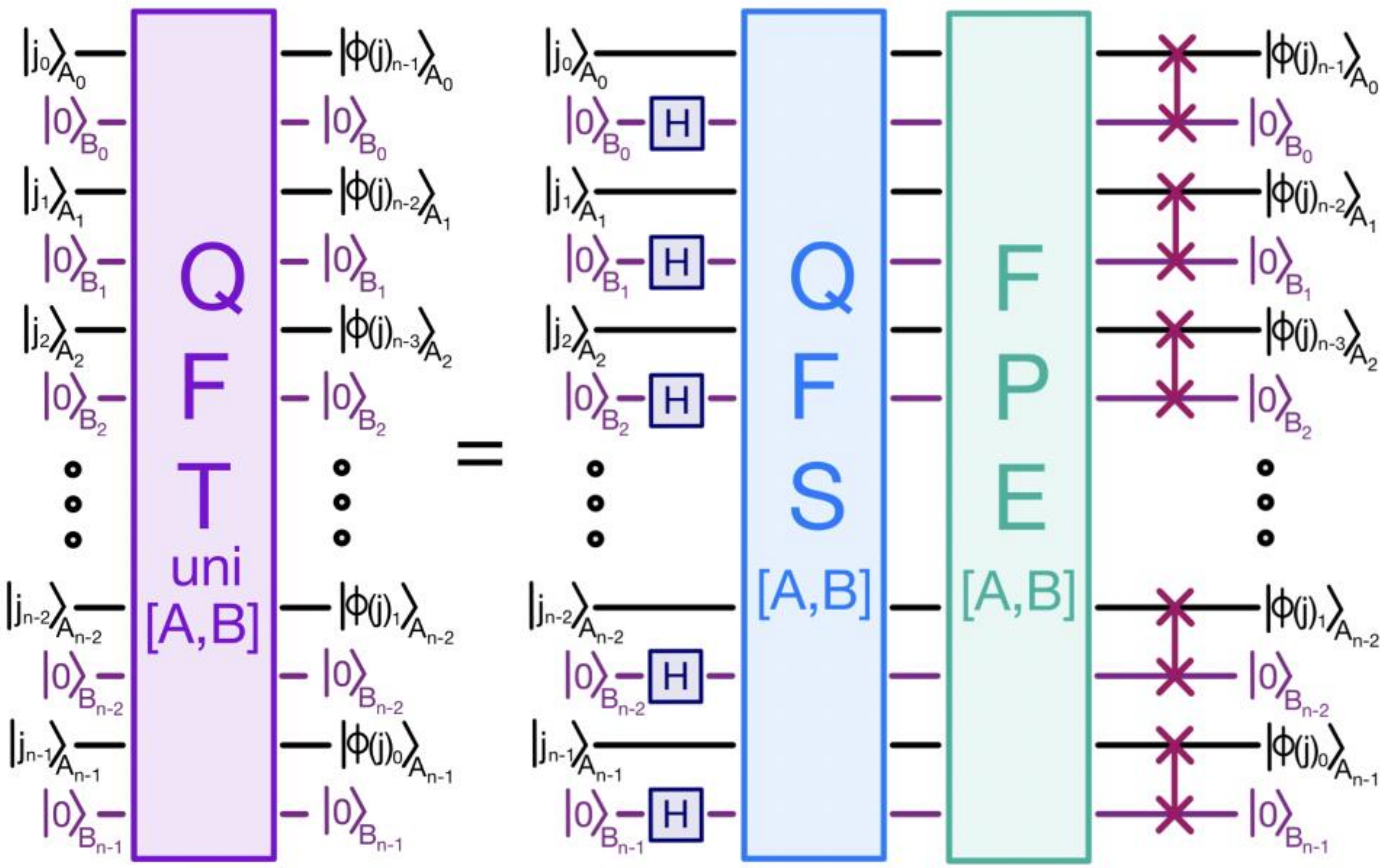
For most inputs, we require only two operations:

i. Quantum Fourier state computation (QFS):

$$|j\rangle|\phi(b)\rangle \rightarrow |j\rangle|\phi(b+j)\rangle$$

ii. Fourier phase estimation (FPE):

$$|b\rangle|\phi(j)\rangle \rightarrow |b \oplus j\rangle|\phi(j)\rangle$$

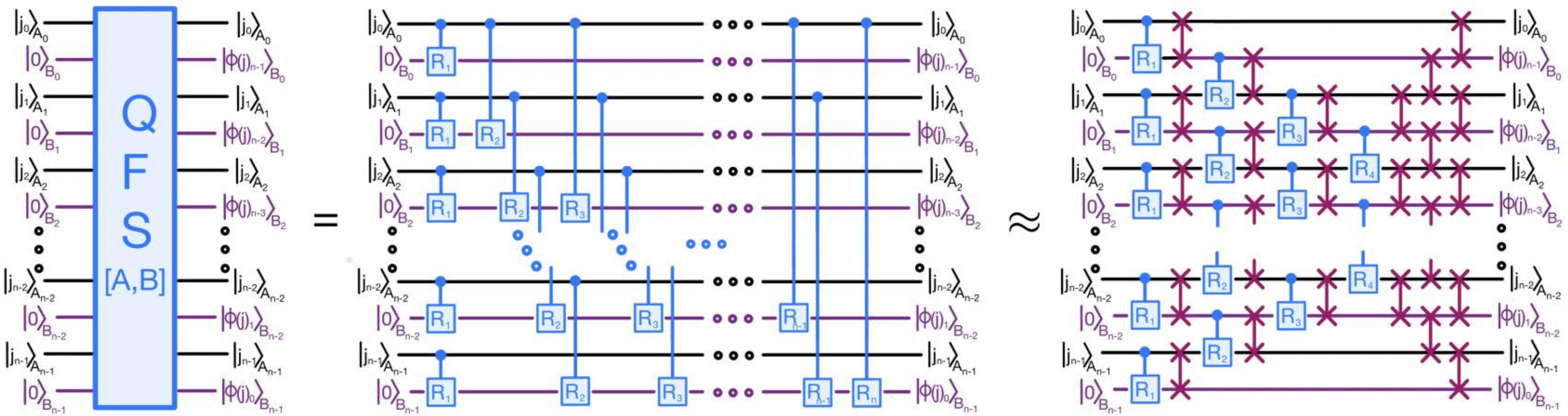


together with a Hadamard transform and SWAP gates:

$$|j\rangle|0\rangle \underbrace{\xrightarrow{H} |j\rangle|\phi(0)\rangle \xrightarrow{\text{QFS}} |j\rangle|\phi(j)\rangle \xrightarrow{\text{FPE}} |0\rangle|\phi(j)\rangle \xrightarrow{\text{SWAP}}}_{\xrightarrow{\text{QFT}_{\text{uni}}}} |\phi(j)\rangle|0\rangle$$

Quantum Fourier state computation (QFS)

arXiv:2504.20832



- Exact QFS: $|j\rangle_A|\phi(b)\rangle_B \overset{\mathrm{QFS}_{AB}}{\rightarrow} |j\rangle_A|\phi(b+j)\rangle_B$
- Approximate by neglecting small rotations, i.e. all phase gates $R_k := \begin{pmatrix} 1 & 0 \\ 0 & e^{2\pi i/2^k} \end{pmatrix}$ for $k > k_{\max} = O(\log \frac{n}{\varepsilon})$

➢ A unitary $\mathrm{QFS}^{(\varepsilon)}$ acting on $\mathcal{S} := \left\{ |\psi\rangle_{ABE} \in \mathcal{H}_{ABE} : |\psi\rangle_{ABE} = \sum_{j=0}^{N-1} \sum_{m=0}^{M-1} \alpha_{j,m} |j\rangle_A |0\rangle_B |m\rangle_E \right\}$

with $\mathrm{dist}_{\mathcal{S}}(\mathrm{QFS}_{AB}, \mathrm{QFS}^{(\varepsilon)}_{AB}) \leq \varepsilon$ can be implemented on a line of $2n$ qubits with nearest-neighbor connectivity with depth $O(\log \frac{n}{\varepsilon})$

arXiv:2504.20832

Fourier phase estimation (FPE)

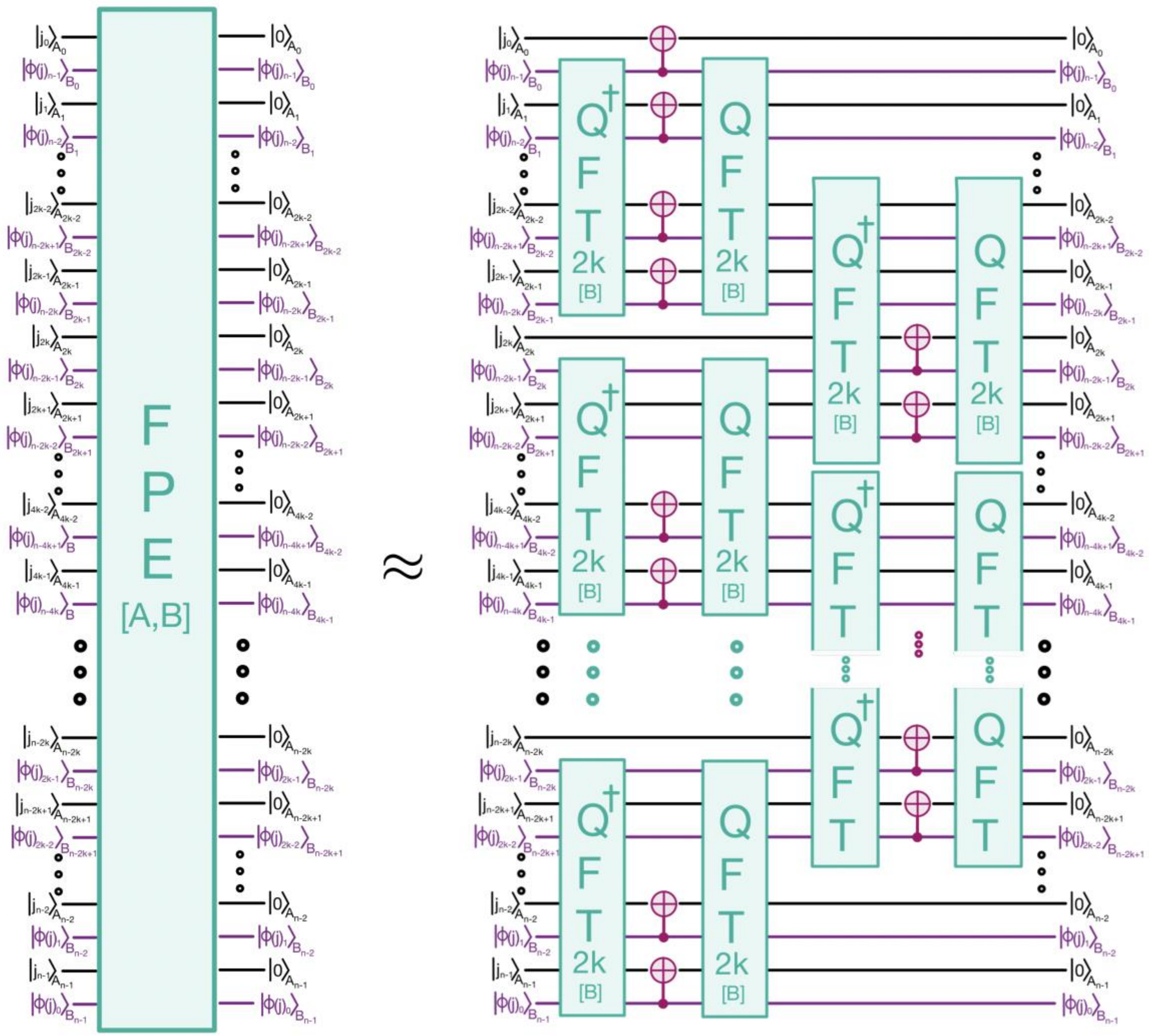
- Exact FPE: $|b\rangle_A |\phi(j)\rangle_B \xrightarrow{\mathrm{FPE}_{AB}} |b \oplus j\rangle_A |\phi(j)\rangle_B$
- Approximation: estimate $|j\rangle$ by small, but exact QFTs that are applied in parallel on $2k$ qubits each, where $k = \mathcal{O}(\log n)$

➢ A unitary $\mathrm{FPE}^{(\varepsilon)}$ with $\mathrm{dist}_{\mathcal{T}^{(p,q)}_{\mathrm{uni}}}(\mathrm{FPE}_{AB}, \mathrm{FPE}^{(\varepsilon)}_{AB}) \leq \varepsilon$ can be implemented on a line of $2n$ qubits with nearest-neighbor connectivity with depth $O(\log \frac{n}{\varepsilon^2})$

! BUT: $\mathcal{T}^{(p)}_{\mathrm{uni}} := \Big\{ |\psi\rangle_{ABE} \in \mathcal{H}_{ABE} : |\psi\rangle_{ABE} = \sum_{j=0}^{N-1} \sum_{m=0}^{M-1} \beta_{j,m} |j\rangle_A |\phi(j)\rangle_B |m\rangle_E \,,$

$$|\sum_m \beta_{j,m} \beta^*_{\ell,m}| \leq \frac{p(n)}{N} \delta_{j,\ell} \,\forall j,\ell \Big\}$$

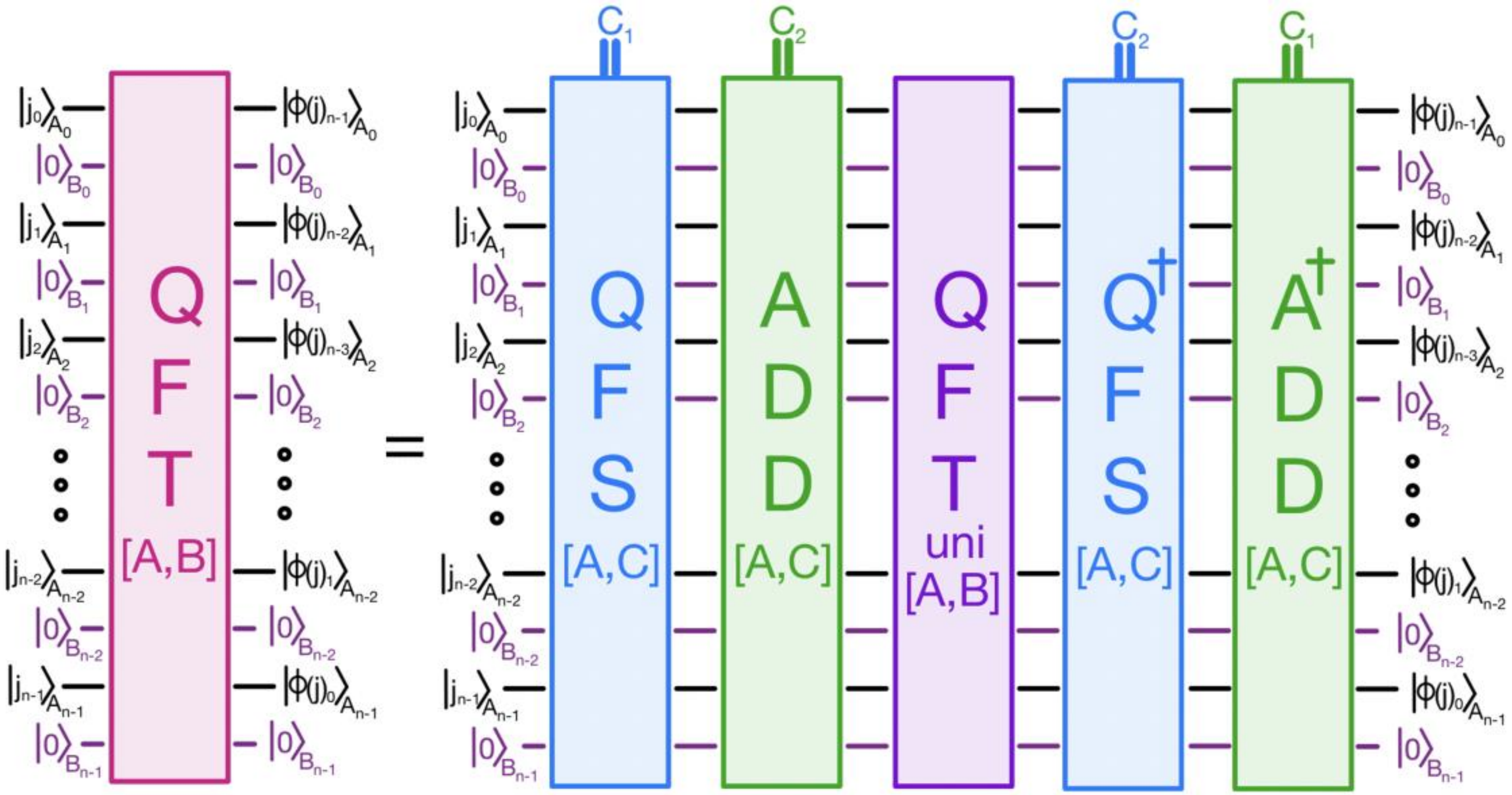
➢ need somewhat uniform inputs



Constructing uniform inputs

arXiv:2504.20832

- Problem: approximate FPE works for most inputs, but not all
- ➢ Constructing a somewhat uniform input is guaranteed to work
- ➢ Requires another operation: Quantum adder (ADD): $|b\rangle|c\rangle \rightarrow |b+c\rangle|c\rangle$
  as well as a classical register C initialized with two randomly chosen numbers $c_1, c_2 \in \{0, \ldots, N-1\}$
- ➢ Constructed an adder that can be implemented on a line of $\mathcal{O}(n)$ qubits with nearest-neighbor connectivity in logarithmic depth using dynamic circuits



arXiv:2504.20832

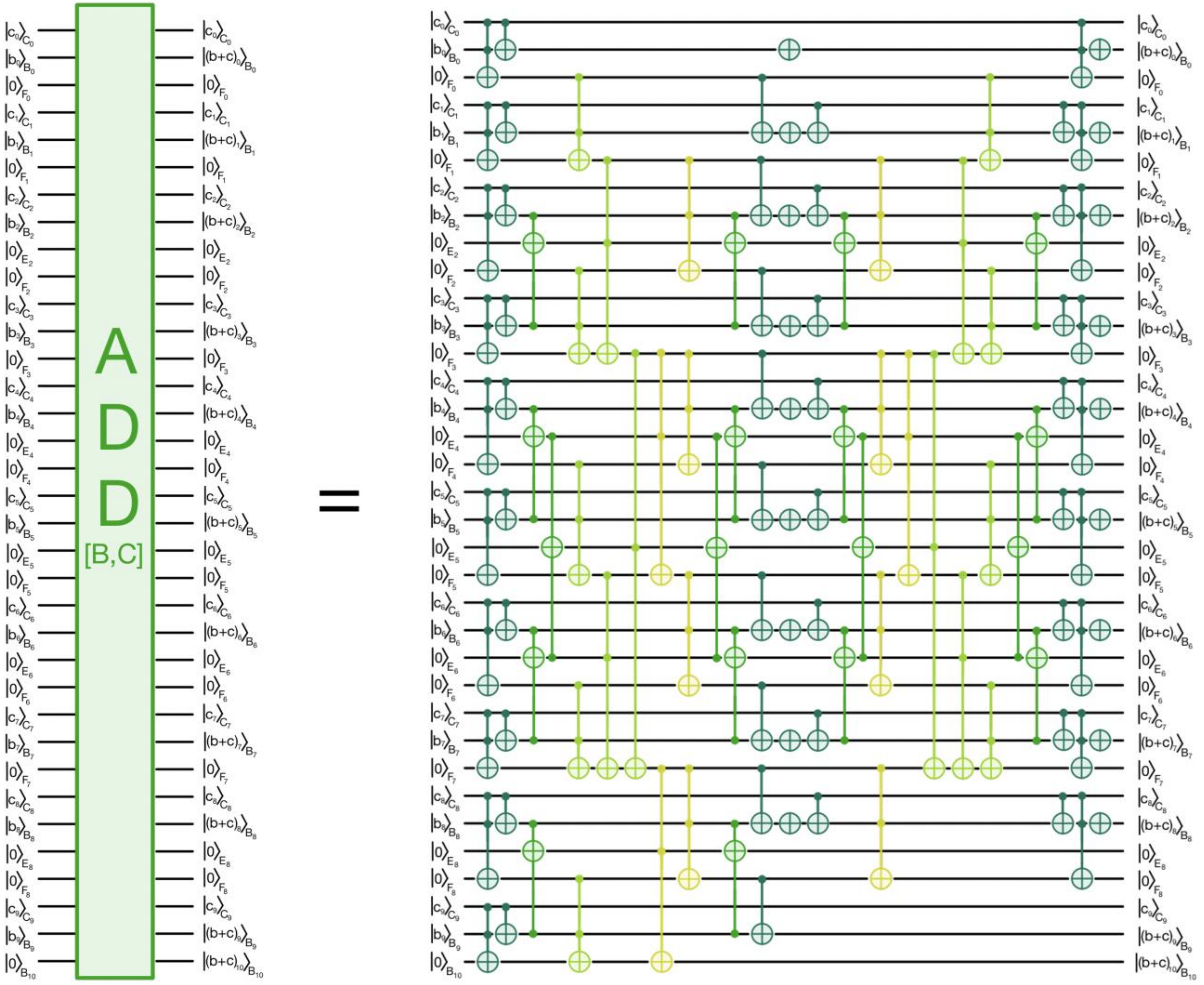
Quantum Adder (ADD)

- ADD: $|b\rangle|c\rangle \xrightarrow{\text{ADD}} |b+c\rangle|c\rangle$

  also equivalent to $|\phi(j)\rangle|\phi(\ell+j)\rangle \rightarrow |\phi(j)\rangle|\phi(\ell)\rangle$
- Draper et al.'s logarithmic-depth quantum carry-lookahead adder seems to require all-to-all connectivity

➢ Notice that in each of the $\mathcal{O}(\log n)$ layers, the long-range gates are not overlapping and thus we can use dynamic circuits to implement them in parallel even on a 1D line

➢ The ADD mapping can be implemented on a line of $5n$ qubits with nearest-neighbor connectivity with depth $\mathcal{O}(\log n)$



T. G. Draper, S. A. Kutin, E. M. Rains, and K. M. Svore, *A logarithmic-depth quantum carry-lookahead adder*

arXiv:2504.20832

Improved Implementation

- **AQFT for uniform inputs:** A unitary $\mathrm{QFT}^{(\varepsilon)}_{\mathrm{uni}}$ acting on uniform states with $\mathrm{dist}_{\mathcal{S}^{(p)}_{\mathrm{uni}}}(\mathrm{QFT}, \mathrm{QFT}^{(\varepsilon)}_{\mathrm{uni}}) \leq \varepsilon$ can be implemented on a line of $2n$ qubits with nearest-neighbor connectivity with depth $O(\log \frac{n}{\varepsilon^2})$
  - ➢ Guaranteed to work for uniform inputs, but also works for most random input states
  - ➢ May be very relevant in practice, as it can always be the first try if the result is verifiable
- **AQFT for general inputs:** A unitary $\mathrm{QFT}^{(\varepsilon)}$ with $\mathrm{dist}_{\mathcal{S}}(\mathrm{QFT}_{AB}, \mathrm{QFT}^{(\varepsilon)}_{AB}) \leq \varepsilon$ can be implemented on a line of $4n$ qubits with nearest-neighbor connectivity with depth $O(\log \frac{n}{\varepsilon^2})$

- Use the principle of deferred measurement on the (ideally) uncorrelated ancilla qubits for uncomputation

➢ Running QFT “backwards” allows to implement the QFS exactly in constant depth

- Potential optimization: recursively replace small QFTs in the FPE by this approximate QFT to reduce depth of FPE to $\mathcal{O}(\log\log n)$ → analysis of errors and trade-offs due to larger pre-factors required

arXiv:2504.20832

Conclusion & Outlook

- Dynamic circuits can help **overcome connectivity constraints**, **reduce the depth** and thereby **drastically improve the compilation of quantum algorithms**
- Their effect will become more significant as we scale up the number of qubits
- Already now we can see the benefit of leveraging dynamic circuits in optimizing the compilation of some quantum algorithms, many more applications to explore!
  - ➢ Improved compilation to overcome current limitations of hardware
  - ➢ Optimizing the resources required in the fault-tolerant compilation

Thank you for your attention!