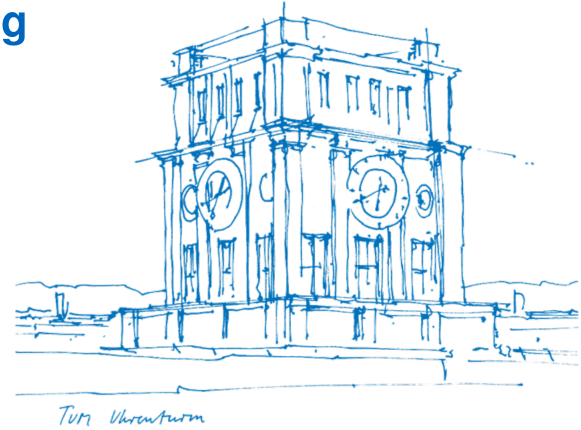
Randomized Benchmarking



Nathaniel Tornow

Seminar: Advanced Topics of Quantum Computing
School of Computation, Information and Technology
Technical University of Munich

December 15th, 2023

The Challenge of Heterogeneity

Remarkable progress in quantum processing unit (QPU) manufacturing

IBM Q IONQ QUANTINUUM AQT

The Challenge of Heterogeneity

Remarkable progress in quantum processing unit (QPU) manufacturing

Challenge: Heterogeneity on two levels:

The Challenge of Heterogeneity

Remarkable progress in quantum processing unit (QPU) manufacturing

Challenge: Heterogeneity on two levels:

1. **Gate-level heterogeneity**
   - Significant **noise variance** of individual physical qubits

The Challenge of Heterogeneity

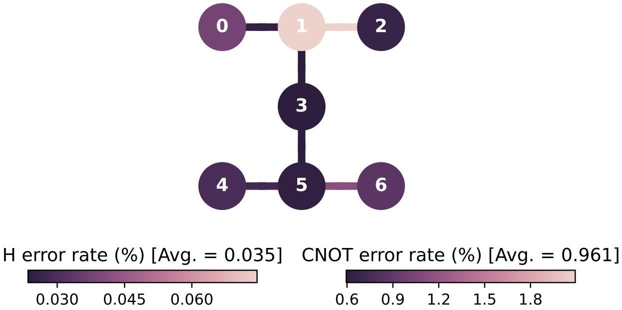
Remarkable progress in quantum processing unit (QPU) manufacturing

Challenge: Heterogeneity on two levels:

1. **Gate-level heterogeneity**
   - Significant **noise variance** of individual physical qubits
2. **QPU-level heterogeneity**
   - **Vastly different properties** of QPU technologies

Challenge #1: Gate-Level Heterogeneity

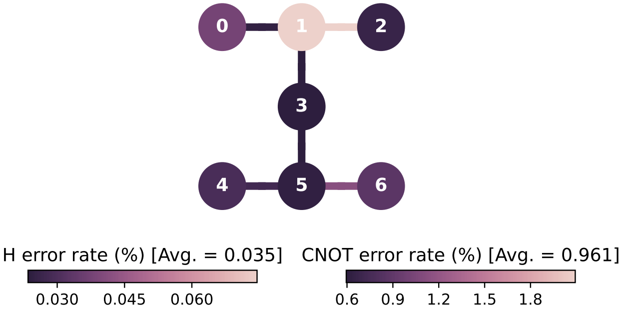
- Gate noise differs significantly between qubits



Gate errors on IBM Perth

Challenge #1: Gate-Level Heterogeneity

- Gate noise differs significantly between qubits



Gate errors on IBM Perth

How can we characterize the errors of gates on individual qubits?

Challenge #2: QPU-level Heterogeneity

Property	IBM Kolkata [1]	IonQ Aria [2]	IBM vs. IonQ

Challenge #2: QPU-level Heterogeneity

Property	IBM Kolkata [1]	IonQ Aria [2]	IBM vs. IonQ
Physical qubits	27 qubits	21 qubits	~ 1.3

Challenge #2: QPU-level Heterogeneity

Property	IBM Kolkata [1]	IonQ Aria [2]	IBM vs. IonQ
Physical qubits	27 qubits	21 qubits	~ 1.3
Errors (1Q / 2Q)	$\sim 2 \cdot 10^{-4}$ / $\sim 9 \cdot 10^{-3}$	$\sim 5 \cdot 10^{-4}$ / $\sim 4 \cdot 10^{-3}$	~ 1

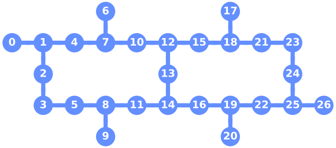
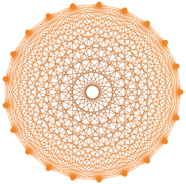
Challenge #2: QPU-level Heterogeneity

Property	IBM Kolkata [1]	IonQ Aria [2]	IBM vs. IonQ
Physical qubits	27 qubits	21 qubits	~ 1.3
Errors (1Q / 2Q)	$\sim 2 \cdot 10^{-4}$ / $\sim 9 \cdot 10^{-3}$	$\sim 5 \cdot 10^{-4}$ / $\sim 4 \cdot 10^{-3}$	~ 1
Duration (1Q / 2Q)	35 ns / 300 ns	135 µs / 600 µs	$\sim 0.01\%$

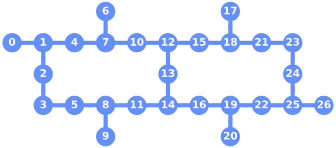
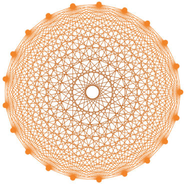
Challenge #2: QPU-level Heterogeneity

Property	IBM Kolkata [1]	IonQ Aria [2]	IBM vs. IonQ
Physical qubits	27 qubits	21 qubits	~ 1.3
Errors (1Q / 2Q)	$\sim 2 \cdot 10^{-4}$ / $\sim 9 \cdot 10^{-3}$	$\sim 5 \cdot 10^{-4}$ / $\sim 4 \cdot 10^{-3}$	~ 1
Duration (1Q / 2Q)	35 ns / 300 ns	135 µs / 600 µs	$\sim 0.01\%$
T1 / T2	100 µs / 600 µs	$10 - 100$ s / ~ 1 s	$\sim 10000\times$

Challenge #2: QPU-level Heterogeneity

Property	IBM Kolkata [1]	IonQ Aria [2]	IBM vs. IonQ
Physical qubits	27 qubits	21 qubits	~ 1.3
Errors (1Q / 2Q)	$\sim 2 \cdot 10^{-4}$ / $\sim 9 \cdot 10^{-3}$	$\sim 5 \cdot 10^{-4}$ / $\sim 4 \cdot 10^{-3}$	~ 1
Duration (1Q / 2Q)	$35\,\text{ns}$ / $300\,\text{ns}$	$135\,\mu\text{s}$ / $600\,\mu\text{s}$	$\sim 0.01\%$
T1 / T2	$100\,\mu\text{s}$ / $600\,\mu\text{s}$	$10 - 100\,\text{s}$ / $\sim 1\,\text{s}$	$\sim 10000\times$
Qubit connectivity			Full

Challenge #2: QPU-level Heterogeneity

Property	IBM Kolkata [1]	IonQ Aria [2]	IBM vs. IonQ
Physical qubits	27 qubits	21 qubits	~ 1.3
Errors (1Q / 2Q)	$\sim 2 \cdot 10^{-4}$ / $\sim 9 \cdot 10^{-3}$	$\sim 5 \cdot 10^{-4}$ / $\sim 4 \cdot 10^{-3}$	~ 1
Duration (1Q / 2Q)	35 ns / 300 ns	135 µs / 600 µs	$\sim 0.01\%$
T1 / T2	100 µs / 600 µs	$10 - 100$ s / ~ 1 s	$\sim 10000\times$
Qubit connectivity			Full

How can we compare different QPU technologies?

Randomized Benchmarking

1. Determine the gate errors of individual qubits or qubit connections?
   $\rightarrow$ **Randomized Benchmarking**

Randomized Benchmarking

1. Determine the gate errors of individual qubits or qubit connections?
   $\rightarrow$ **Randomized Benchmarking**
2. Measure and compare the performance of entire QPU systems?
   $\rightarrow$ **Quantum Volume Protocol**

Outline

1. Gate-Level Characterization (Randomized Benchmarking)
   - Standard Randomized Benchmarking
   - Interleaved Randomized Benchmarking

2. QPU-level Randomized Benchmarks

Randomized Benchmarking Idea

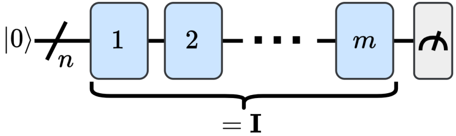
Goal: Measure the error of gates on specific physical qubits/ qubit connections

Randomized Benchmarking Idea

Goal: Measure the error of gates on specific physical qubits/ qubit connections

Idea:

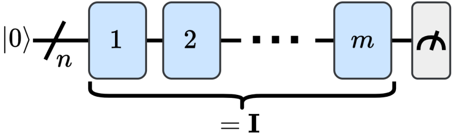
- Run sequences of m *random* gates on n qubits, that overall compose the identity gate $\mathbf{I}$
- Without errors, the probability of measuring $p(|0\rangle)$ should be 1.0 (success)
- However, more gates add more noise, decaying $p(|0\rangle)$ to $\frac{1}{2^n}$

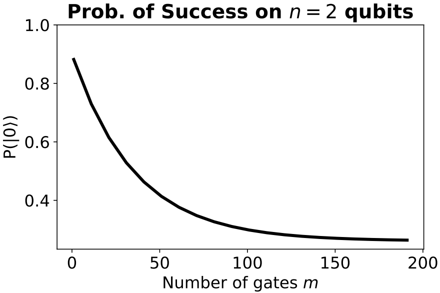


Randomized Benchmarking Idea

Goal: Measure the error of gates on specific physical qubits/ qubit connections
Idea:

- Run sequences of m *random* gates on n qubits, that overall compose the identity gate $\mathbf{I}$
- Without errors, the probability of measuring $p(|0\rangle)$ should be 1.0 (success)
- However, more gates add more noise, decaying $p(|0\rangle)$ to $\frac{1}{2^n}$





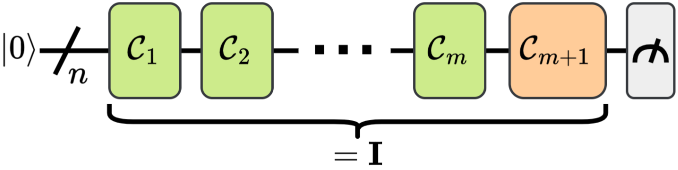
Random Clifford Sequences

Problem: Efficiently create gate-sequences that compose to the $\mathbf{I}$ gate

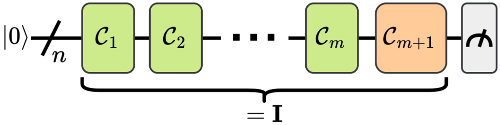
Random Clifford Sequences

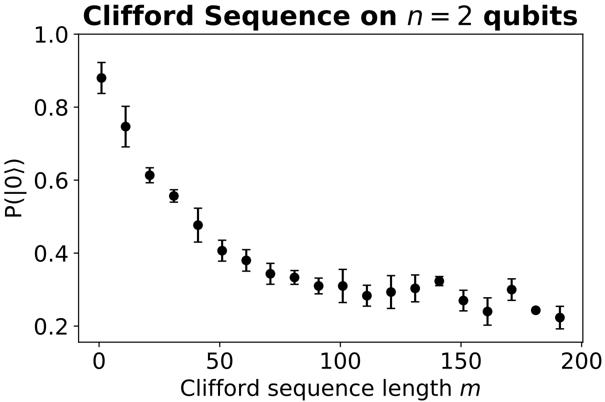
Problem: Efficiently create gate-sequences that compose to the $\mathbf{I}$ gate

- Construct sequences of m random Clifford gates $\mathcal{C}_i$ [3]
- Efficient computation of the inverse of large sequences
- Append the inverse $\mathcal{C}_{m+1}$ to get the identity in total



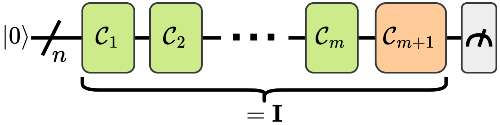
Running Random Clifford Sequences

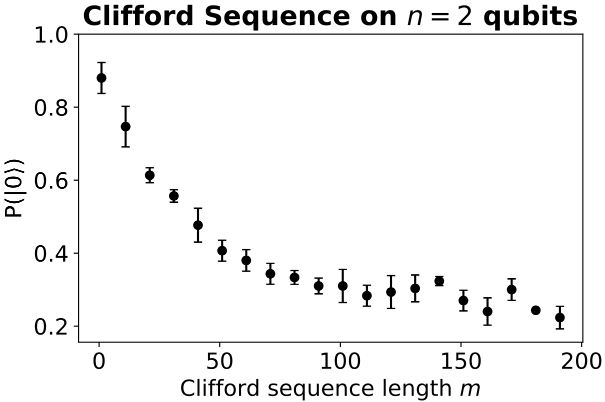




Random Clifford Sequences qubits [4, 5] of IBM Nairobi (7-qubit device)

Running Random Clifford Sequences





Random Clifford Sequences qubits [4, 5] of IBM Nairobi (7-qubit device)

How can we determine the error per Clifford (EPC)?

Randomized Benchmarking Intuition

- Depolarizing quantum channel:

$$\rho_f = \alpha \rho_i + \frac{1-\alpha}{2^n}\,\mathbf{I}$$

Randomized Benchmarking Intuition

- Depolarizing quantum channel:

$$\rho_f = \alpha \rho_i + \frac{1-\alpha}{2^n}\mathbf{I}$$

- If we run m random Clifford gates, the channel can be described as

$$\rho_f^m = \alpha^m \rho_i + \frac{1-\alpha^m}{2^n}\mathbf{I}$$

Randomized Benchmarking Intuition

- Depolarizing quantum channel:

$$\rho_f = \alpha \rho_i + \frac{1-\alpha}{2^n}\,\mathbf{I}$$

- If we run m random Clifford gates, the channel can be described as

$$\rho_f^m = \alpha^m \rho_i + \frac{1-\alpha^m}{2^n}\,\mathbf{I}$$

- We can derive the probability of success after the Clifford sequence:

$$p(|0\rangle) = \alpha^m + \frac{1-\alpha^m}{2^n} = \frac{2^n-1}{2^n}\alpha^m + \frac{1}{2^n} = A_0\alpha^m + B_0$$

Randomized Benchmarking Intuition

- Depolarizing quantum channel:

$$\rho_f = \alpha\rho_i + \frac{1-\alpha}{2^n}\,\mathbf{I}$$

- If we run m random Clifford gates, the channel can be described as

$$\rho_f^m = \alpha^m\rho_i + \frac{1-\alpha^m}{2^n}\,\mathbf{I}$$

- We can derive the probability of success after the Clifford sequence:

$$p(|0\rangle) = \alpha^m + \frac{1-\alpha^m}{2^n} = \frac{2^n-1}{2^n}\alpha^m + \frac{1}{2^n} = A_0\alpha^m + B_0$$

- Therefore, the error-rate per Clifford (EPC) is:

$$r = 1 - \left(\alpha + \frac{1-\alpha}{2^n}\right)$$

Randomized Benchmarking Intuition

- Depolarizing quantum channel:

$$\rho_f = \alpha\rho_i + \frac{1-\alpha}{2^n}\,\mathbf{I}$$

- If we run m random Clifford gates, the channel can be described as

$$\rho_f^m = \alpha^m\rho_i + \frac{1-\alpha^m}{2^n}\,\mathbf{I}$$

- We can derive the probability of success after the Clifford sequence:

$$p(|0\rangle) = \alpha^m + \frac{1-\alpha^m}{2^n} = \frac{2^n-1}{2^n}\alpha^m + \frac{1}{2^n} = A_0\alpha^m + B_0$$

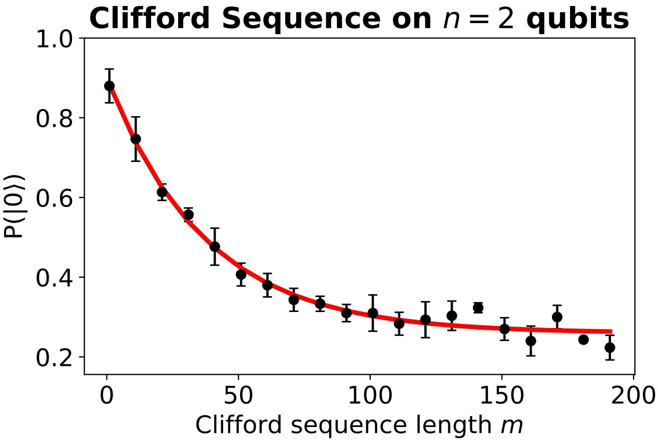
- Therefore, the error-rate per Clifford (EPC) is:

$$r = 1 - \left(\alpha + \frac{1-\alpha}{2^n}\right)$$

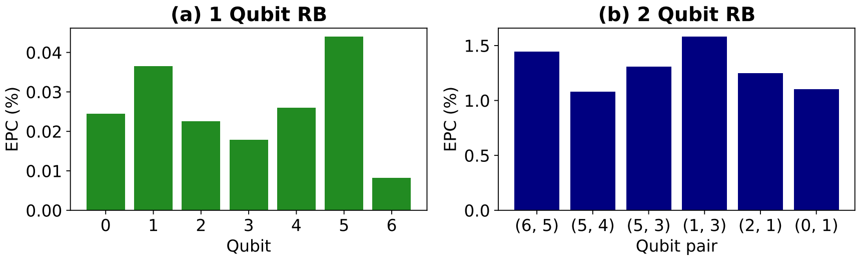
- Use curve-fitting to find α

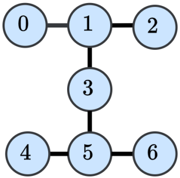
Curve-fit to find the Error per Clifford

$$p(|0\rangle) = A_0 \alpha^m + B_0 \implies \alpha = 0.973 \implies r = 1 - \left(\alpha + \frac{1-\alpha}{2^n}\right) = 0.019$$



Randomized Benchmarking on IBM Nairobi





IBM Nairobi

Interleaved Randomized Benchmarking [4]

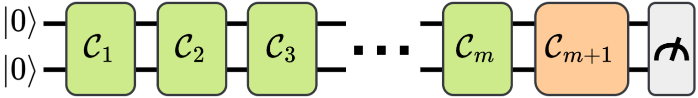
Problem: How can we determine the error of specific QPU-native gates (e.g., CNOT gate)?

Interleaved Randomized Benchmarking [4]

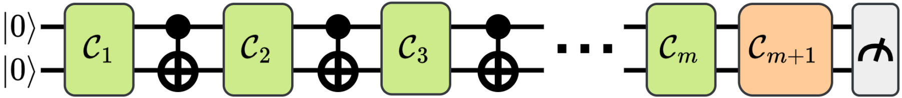
Problem: How can we determine the error of specific QPU-native gates (e.g., CNOT gate)?

Idea:

- Interleave a specific gate between the Clifford gates
- Measure the relative error to the standard EPC

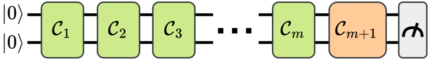


(a) Standard Clifford sequence

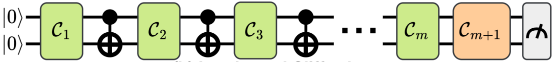


(b) Interleaved Clifford sequence

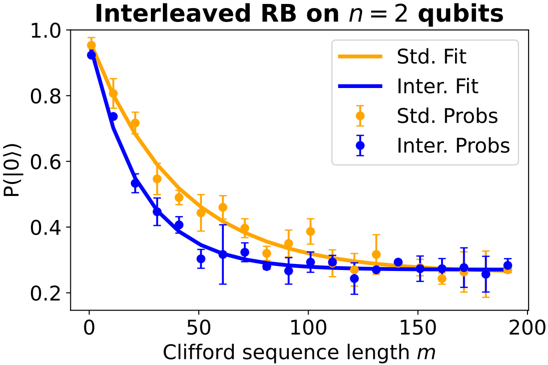
Interleaved Randomized Benchmarking Curve-Fitting



(a) Standard Clifford sequence



(b) Interleaved Clifford sequence



Error per Gate

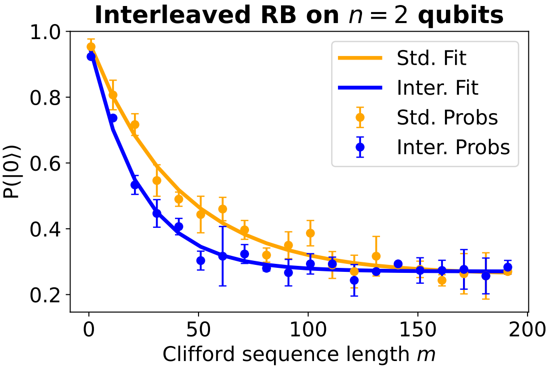
- With the interleaved RB, we get α_{std} and α_{inter} from fitting both to $p(|0\rangle) = A_0\alpha^m + B_0$

Error per Gate

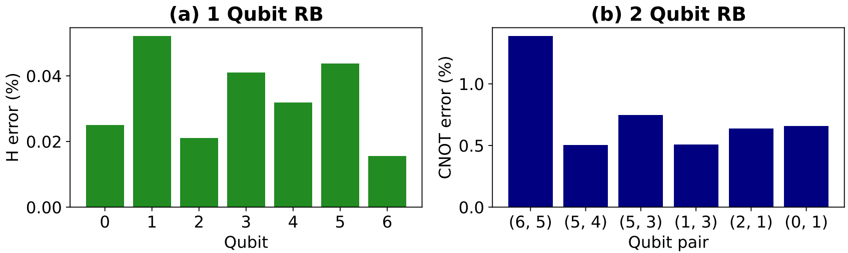
- With the interleaved RB, we get α_{std} and α_{inter} from fitting both to $p(|0\rangle) = A_0\alpha^m + B_0$
- Define the relative decay rate as $\alpha_{rel} = \alpha_{std}/\alpha_{inter}$

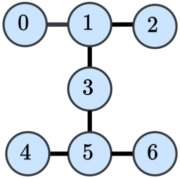
Error per Gate

- With the interleaved RB, we get α_{std} and α_{inter} from fitting both to $p(|0\rangle) = A_0\alpha^m + B_0$
- Define the relative decay rate as $\alpha_{rel} = \alpha_{std}/\alpha_{inter}$
- The error per gate is then defined as $r_{gate} = 1 - \left(\alpha_{rel} + \frac{1-\alpha_{rel}}{2^n}\right) \approx 0.008$



Error per gate on IBM Perth





IBM Nairobi

Outline

Performance of QPU-systems

Challenge:

- Measure and compare the performance of QPUs
- Gate-level RB insufficient: no measure of qubit-connectivity, crosstalk, etc.

Performance of QPU-systems

Challenge:

- Measure and compare the performance of QPUs
- Gate-level RB insufficient: no measure of qubit-connectivity, crosstalk, etc.

Goal:

- General benchmark to measure the performance of an entire QPU

Performance of QPU-systems

Challenge:

- Measure and compare the performance of QPUs
- Gate-level RB insufficient: no measure of qubit-connectivity, crosstalk, etc.

Goal:

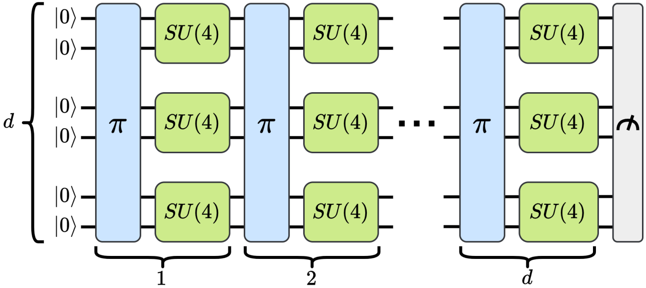
- General benchmark to measure the performance of an entire QPU

Quantum Volume (QV) [5]: A single-number metric with a standardized protocol

Determines the largest random square circuit a QPU can successfully run

The Quantum Volume (QV) model circuit

- **Model circuit:** d layers of random permutations π and random $SU(4)$ gates



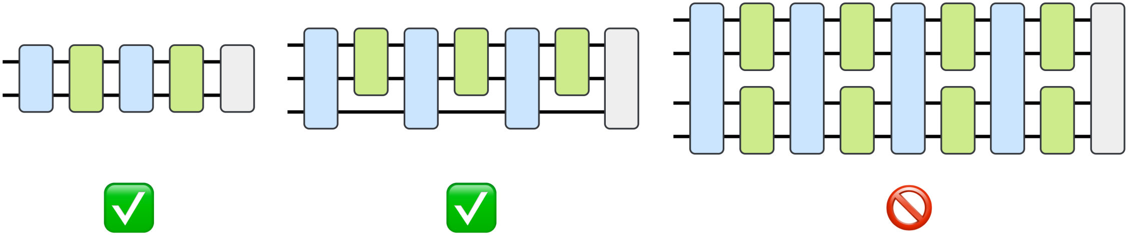
- *Random* circuit, represents a general circuit used in algorithms
- Highly connected, required high-fidelity two-qubit gates

The Quantum Volume Protocol

- A QPU has a $QV = 2^d$ if it successfully implements a QV circuit of depth d.

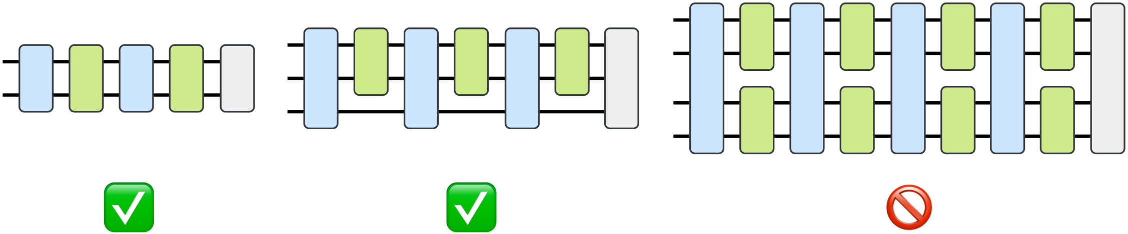
The Quantum Volume Protocol

- A QPU has a $QV = 2^d$ if it successfully implements a QV circuit of depth d.



The Quantum Volume Protocol

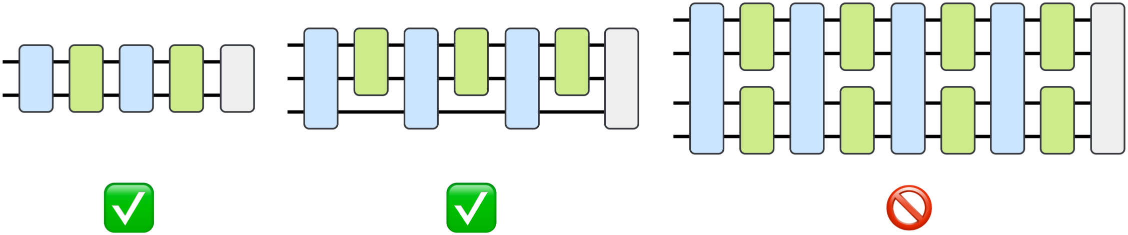
- A QPU has a $QV = 2^d$ if it successfully implements a QV circuit of depth d.



- QPU can successfully implement QV-circuits up to $d = 3$
- Therefore, $QV = 2^d = 8$
- Corresponds to the classical simulation-cost

The Quantum Volume Protocol

- A QPU has a $QV = 2^d$ if it successfully implements a QV circuit of depth d.



- QPU can successfully implement QV-circuits up to $d = 3$
- Therefore, $QV = 2^d = 8$
- Corresponds to the classical simulation-cost
- **What is a successful circuit implementation?**

Successful Circuit Implementation

- Derived from the **heavy output generation problem**

Successful Circuit Implementation

- Derived from the **heavy output generation problem**
- **Successful circuit implementation**:
  If the QPU produces a set of outputs x, such that the **heavy output probability (HOP)** is at least $2/3$

Successful Circuit Implementation

- Derived from the **heavy output generation problem**
- **Successful circuit implementation**:
  If the QPU produces a set of outputs x, such that the **heavy output probability (HOP)** is at least $2/3$
- **Example:**
  - QV circuit U

$$P_U = \{\text{“}\mathbf{00}\text{”} : 0.3,\ \text{“}\mathbf{01}\text{”} : 0.2,\ \text{“}\mathbf{10}\text{”} : 0.0,\ \text{“}\mathbf{11}\text{”} : 0.5\},\ p_{med} = 0.25$$

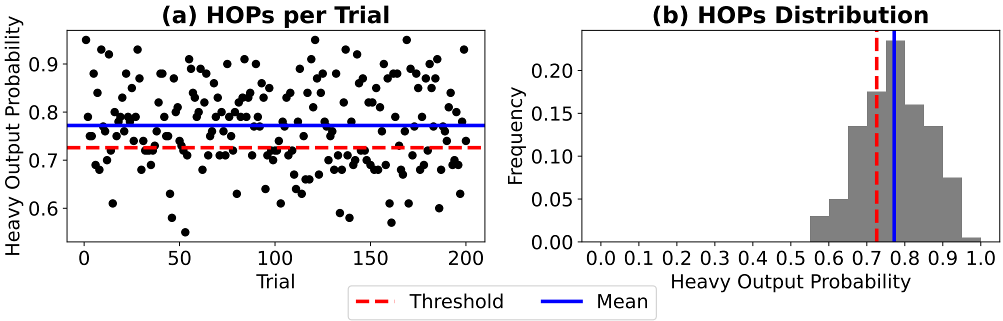
$$H_U = \{\text{“}\mathbf{00}\text{”}, \text{“}\mathbf{11}\text{”}\}$$

- Noisy circuit U' on QPU.
- Successful, if $HOP = p_{U'}(\text{“}\mathbf{00}\text{”}) + p_{U'}(\text{“}\mathbf{11}\text{”}) \geq 2/3$

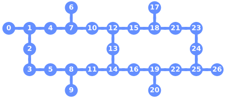
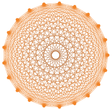
Successful Circuit Implementation (2)

- We want to be **confident** that the QPU is successful for a given depth d
- Run n trials and get heavy output probability p_{mean}
- 97.5% confidence if $p_{mean} \geq 2/3 + 2\sigma$

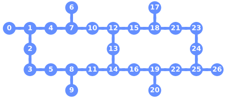
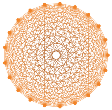
QV experiment with $d = 3$ on IBM Kolkata (simulated)



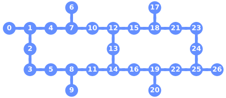
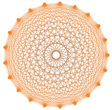
Quantum Volume: IBMQ Kolkata vs. IonQ Aria

Property	**IBM Kolkata**	**IonQ Aria**
Errors (1Q / 2Q)	$\sim 2 \cdot 10^{-3}$ / $\sim 9 \cdot 10^{-4}$	$\sim 5 \cdot 10^{-3}$ / $\sim 4 \cdot 10^{-4}$
T1 / T2	$100\mu s$ / $70\mu s$	$10 - 100s$ / $\sim 1000ms$
Qubit connectivity		

Quantum Volume: IBMQ Kolkata vs. IonQ Aria

Property	**IBM Kolkata**	**IonQ Aria**
Errors (1Q / 2Q)	$\sim 2 \cdot 10^{-3}$ / $\sim 9 \cdot 10^{-4}$	$\sim 5 \cdot 10^{-3}$ / $\sim 4 \cdot 10^{-4}$
T1 / T2	$100\mu s$ / $70\mu s$	$10 - 100s$ / $\sim 1000ms$
Qubit connectivity		
Quantum Volume	$\mathbf{2^4}$	$\mathbf{2^4}$

Quantum Volume: IBMQ Kolkata vs. IonQ Aria

Property	IBM Kolkata	IonQ Aria
Errors (1Q / 2Q)	$\sim 2 \cdot 10^{-3}$ / $\sim 9 \cdot 10^{-4}$	$\sim 5 \cdot 10^{-3}$ / $\sim 4 \cdot 10^{-4}$
T1 / T2	$100\mu s$ / $70\mu s$	$10 - 100s$ / $\sim 1000ms$
Qubit connectivity		
Quantum Volume	$\mathbf{2^4}$	$\mathbf{2^4}$

Current QV record: 2^{19} on the 20-qubit H1 QPU (Quantinuum)

Quantum Volume: Outlook

- Quantum Volume protocol has limitations
  - Requires classical simulation to generate heavy output set
  - 2^x-valued (e.g., no values between $QV = 64$ and $QV = 128$ possible)
  - Does not measure the execution time
  - Only measures the “best” qubits of a system

Quantum Volume: Outlook

- Quantum Volume protocol has limitations
  - Requires classical simulation to generate heavy output set
  - 2^x-valued (e.g., no values between $QV = 64$ and $QV = 128$ possible)
  - Does not measure the execution time
  - Only measures the “best” qubits of a system
- **Circuit Layer Operations per Second (CLOPS)** [6]
  - Addition to Quantum Volume
  - Factors in the time to (iteratively) execute circuits

Quantum Volume: Outlook

- Quantum Volume protocol has limitations
  - Requires classical simulation to generate heavy output set
  - 2^x-valued (e.g., no values between $QV = 64$ and $QV = 128$ possible)
  - Does not measure the execution time
  - Only measures the “best” qubits of a system
- **Circuit Layer Operations per Second (CLOPS)** [6]
  - Addition to Quantum Volume
  - Factors in the time to (iteratively) execute circuits
- **Layer Fidelity** [7]
  - Fine-grained fidelity metric $\in [0, 1]$
  - Benchmarks every qubit of a QPU
  - Builds on the RB protocol

Conclusion

Challenge: Heterogeneity on the **gate-level** and **QPU-level**

1. How to characterize gate-errors?
2. How to compare and measure the performance of different QPUs?

Solution: Randomized Benchmarking (RB)

- **(Interleaved) RB** to characterize gate-errors on individual qubits
- **Quantum Volume (QV)** to benchmark the performance of entire QPUs

Check out my implementation: github.com/nathanieltornow/rand_bench

IBMQ.
Ibmq resources.
https://quantum.ibm.com/services/resources, 2023.
Accessed: December 13, 2023.

IonQ.
Ionq aria: Practical performance.
https://ionq.com/resources/ionq-aria-practical-performance, 2023.
Accessed: December 13, 2023.

Easwar Magesan, Jay M Gambetta, and Joseph Emerson.
Scalable and robust randomized benchmarking of quantum processes.
Physical review letters, 106(18):180504, 2011.

Easwar Magesan, Jay M Gambetta, Blake R Johnson, Colm A Ryan, Jerry M Chow, Seth T Merkel, Marcus P Da Silva, George A Keefe, Mary B Rothwell, Thomas A Ohki, et al.
Efficient measurement of quantum gate error by interleaved randomized benchmarking.
Physical review letters, 109(8):080505, 2012.

Andrew W Cross, Lev S Bishop, Sarah Sheldon, Paul D Nation, and Jay M Gambetta.
Validating quantum computers using randomized model circuits.
Physical Review A, 100(3):032328, 2019.

Andrew Wack, Hanhee Paik, Ali Javadi-Abhari, Petar Jurcevic, Ismael Faro, Jay M Gambetta, and Blake R Johnson.
Quality, speed, and scale: three key attributes to measure the performance of near-term quantum computers.
arXiv preprint arXiv:2110.14108, 2021.

David C McKay, Ian Hincks, Emily J Pritchett, Malcolm Carroll, Luke CG Govia, and Seth T Merkel.
Benchmarking quantum processor performance at scale.
arXiv preprint arXiv:2311.05933, 2023.