Performance Survey of Current Gate-Model Quantum Processors

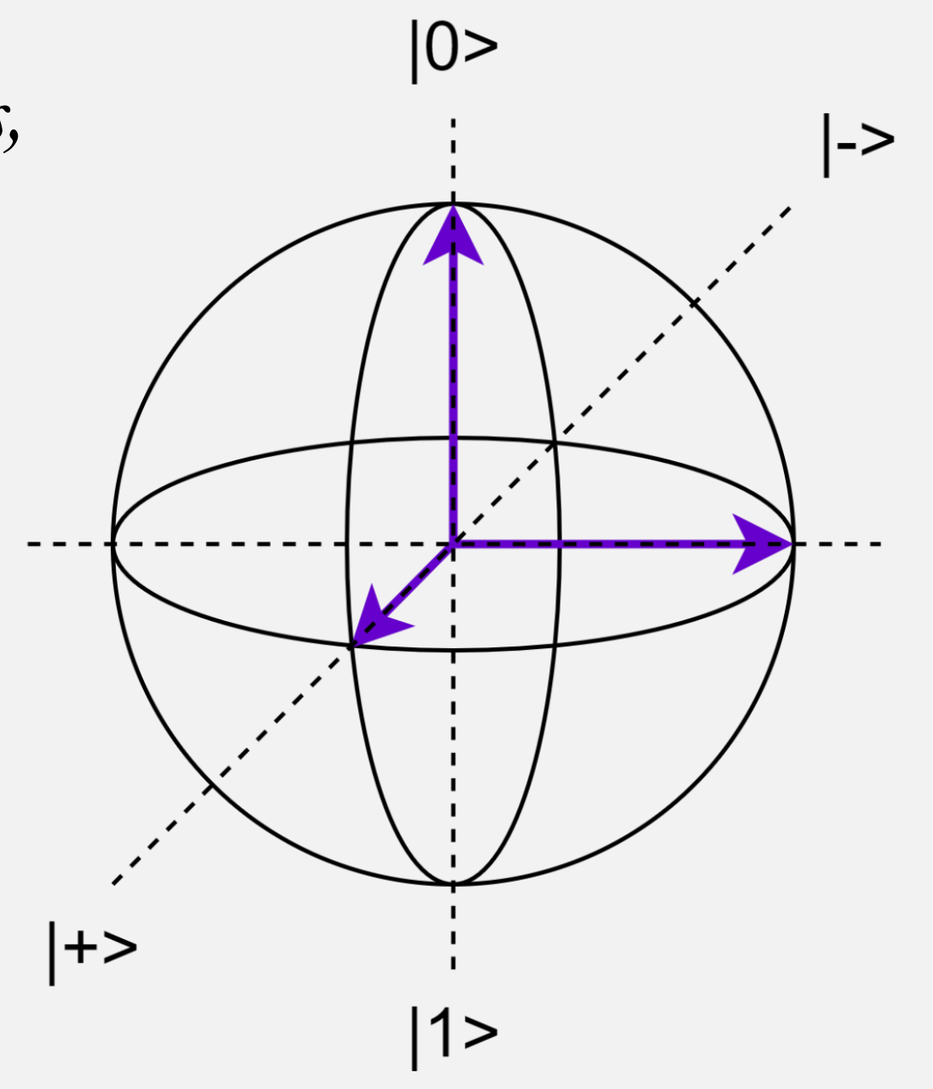
現在のゲート型量子プロセッサの性能調査

Yikai MAO, Masaaki KONDO, Hideharu AMANO
Graduate School of Science and Technology, Keio University

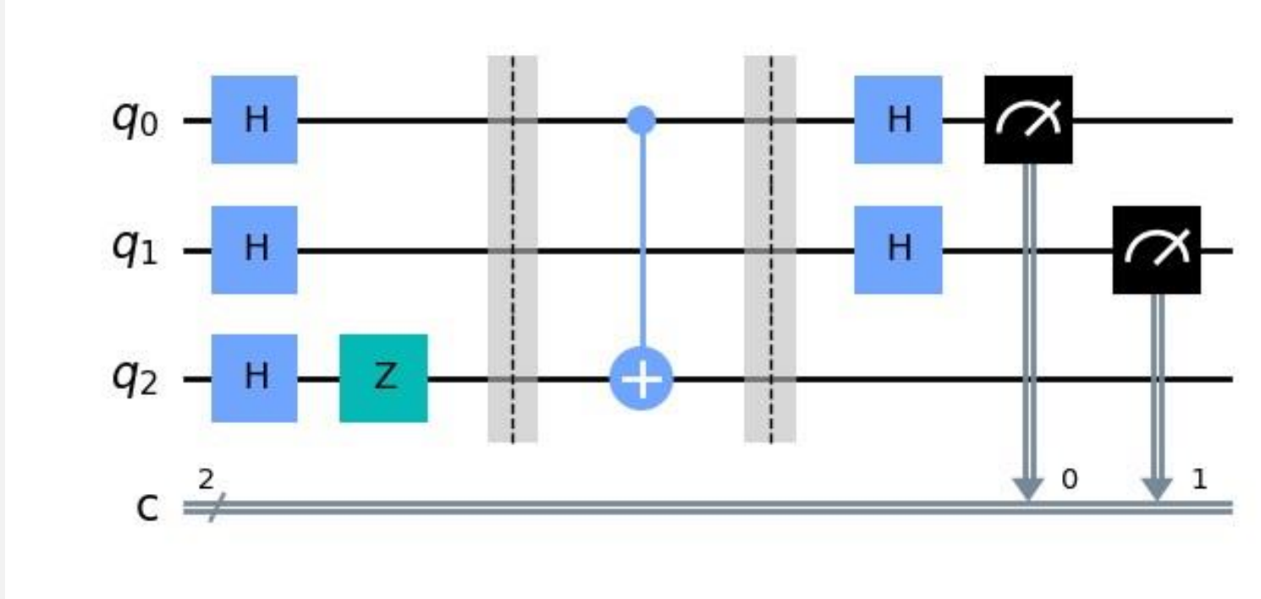
INTRODUCTION

Quantum computing is already in practice. As part of the *Feasibility Studies on Next-Generation Supercomputing Infrastructures,* we want to investigate and study the current status of Quantum Computing, especially the performance of the popular gate-model quantum processors, and the possibility of using its power to augment the next flagship supercomputing system. To summarize:

- Survey current gate-model quantum processors
- Evaluate programmability on different cloud platforms
- Investigate usability with practical quantum algorithms
- Test performance, especially their noisy output fidelity



BACKGROUND



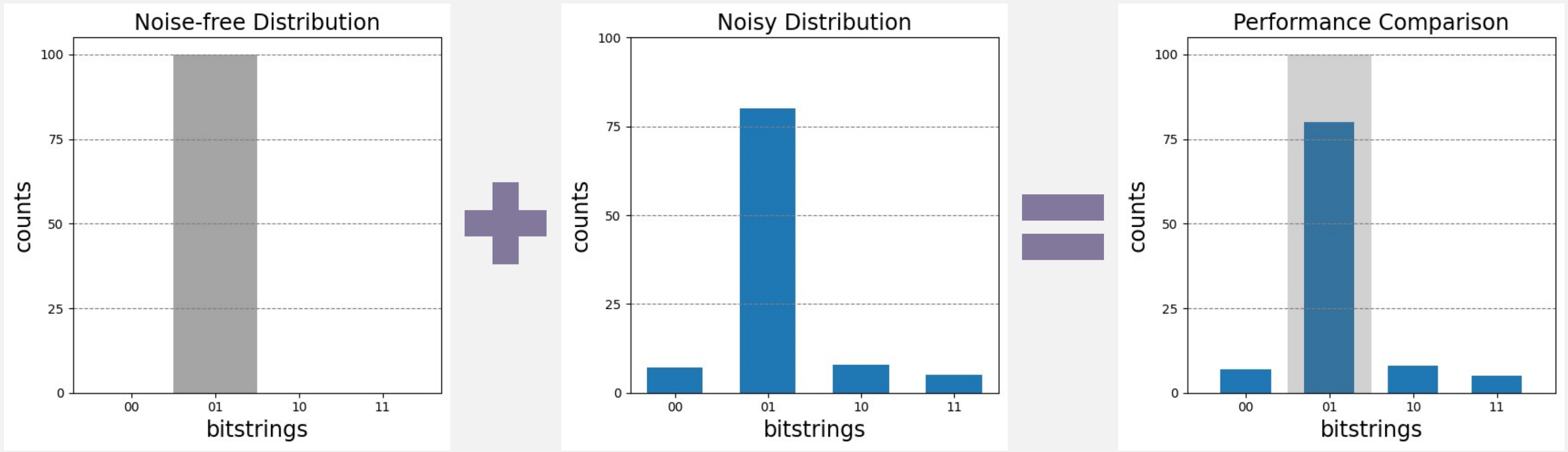
Gate-model quantum computing use quantum circuits to describe the calculation. Inside a quantum circuit, various quantum gates are applied to different qubits. These gates manipulate the qubits in complex computing spaces, using superposition and entanglement to achieve exponential speed-up for traditionally hard tasks, like prime factorization. Currently, the hardware implementation of quantum processors is still imperfect. One of the major problem is that they are easily affected by external noise, causing the output to distribute around the correct answer. To mitigate the problem, quantum circuits are usually executed many times to collect the high-probability output.

METHODOLOGY

We surveyed 9 publicly available gate-model quantum processors on the amazon AWS and the IBM Q cloud platform, listed below:

1. `Harmony` by IonQ, 11 qubit (ion-trap)
2. `Lucy` by Oxford Quantum Circuits, 8 qubit (superconducting)
3. `Aspen-M-3` by Rigetti, 79 qubit (superconducting)
4. `ibm_nairobi` by IBM, 7 qubit (superconducting)
5. `ibm_oslo` by IBM, 7 qubit (superconducting)
6. `ibmq_belem` by IBM, 5 qubit (superconducting)
7. `ibmq_lima` by IBM, 5 qubit (superconducting)
8. `ibmq_manila` by IBM, 5 qubit (superconducting)
9. `ibmq_quito` by IBM, 5 qubit (superconducting)

The test circuits are picked from the `QASMBench` opensource benchmark suite, according to the processor's topology and computation complexity.

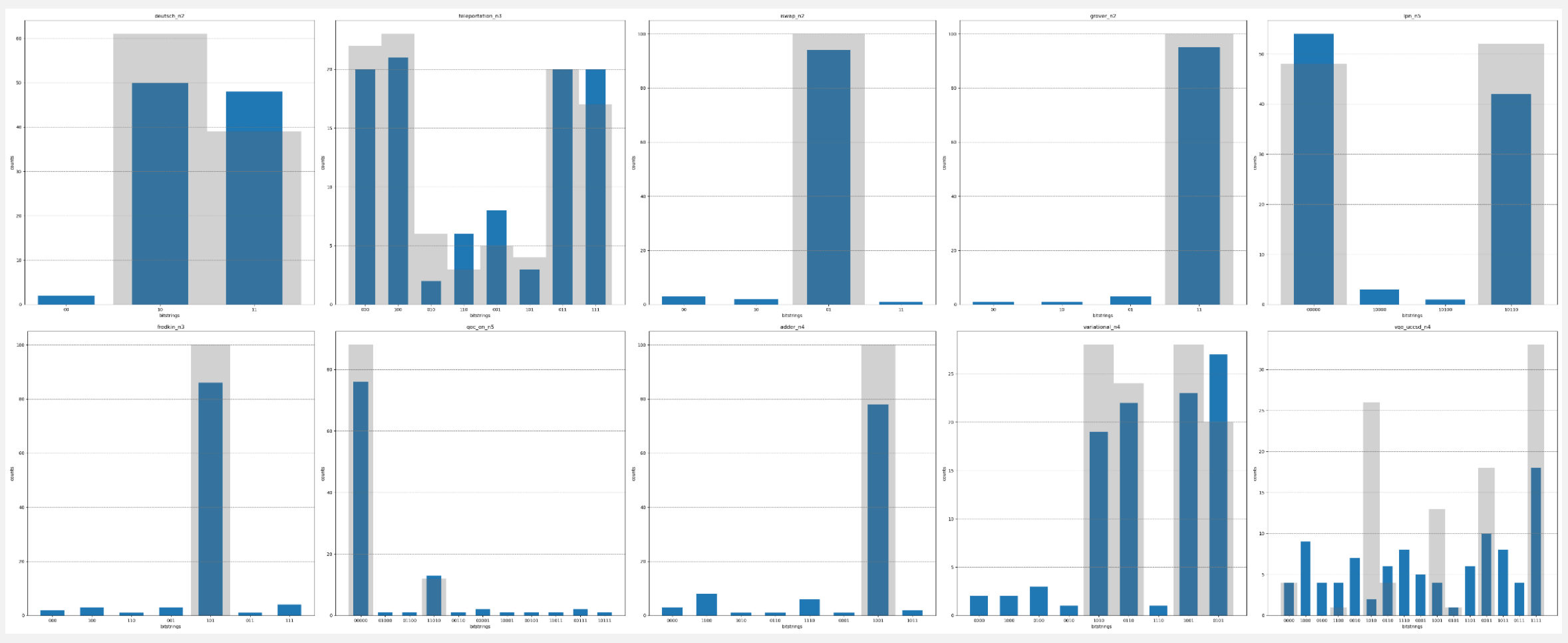


Here is how the figures in the result section are constructed. First, a noise-free simulation of the circuit was performed, generating the expected output distribution in grey. Then the circuit was executed on the real processor, producing a noisy output distribution in blue. The two distributions are merged to give an intuitive comparison of how noise affects the processor's performance.

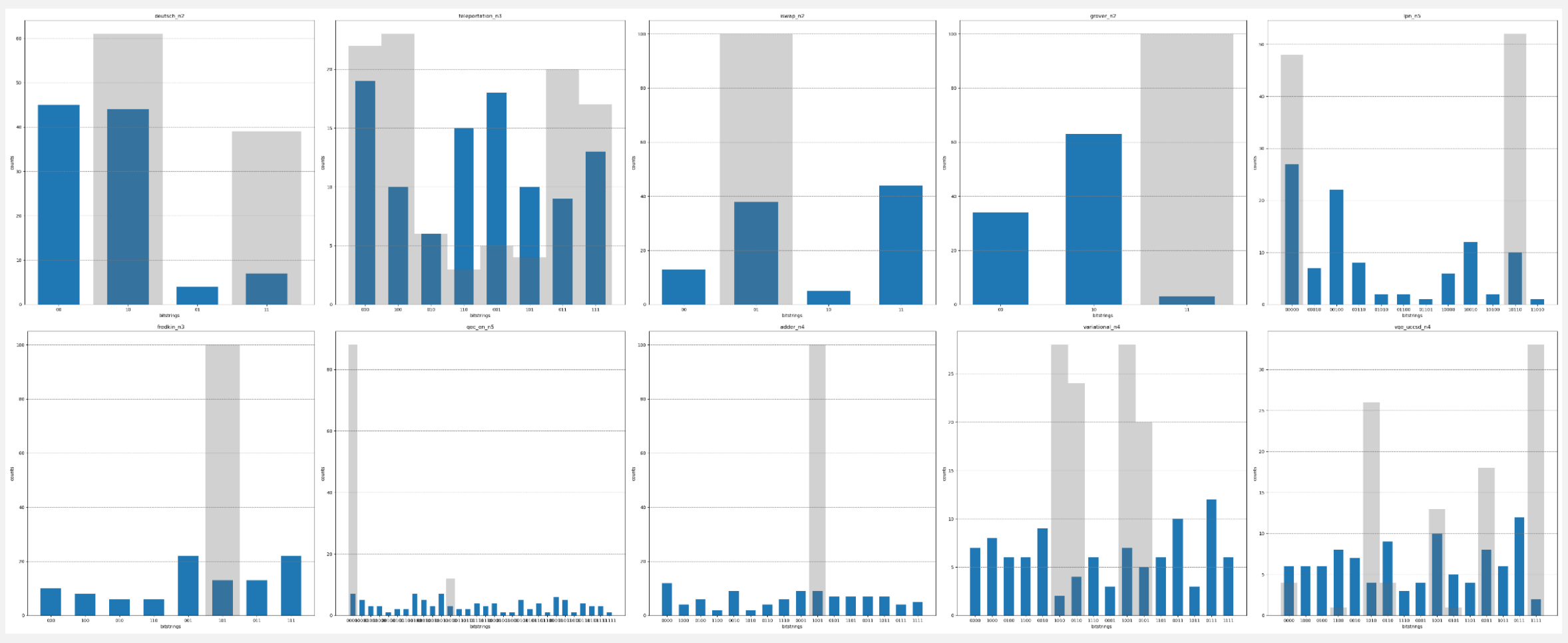
RESULT

Due to space limitations, only the top-5 simplest and most complex quantum circuits are displayed here. The first row are the top-5 simplest algorithms: 1. Deutsch algorithm, 2. Quantum teleportation, 3. entangling swapping gate, 4. Grover's algorithm, 5. Learning parity with noise. The second row are the top-5 most complex algorithms: 1. Controlled-swap gate, 2. Quantum repetition code encoder, 3. Quantum ripple-carry adder, 4. Variational ansatz with a linear-swap network, 5. Variational quantum eigensolver with UCCSD.

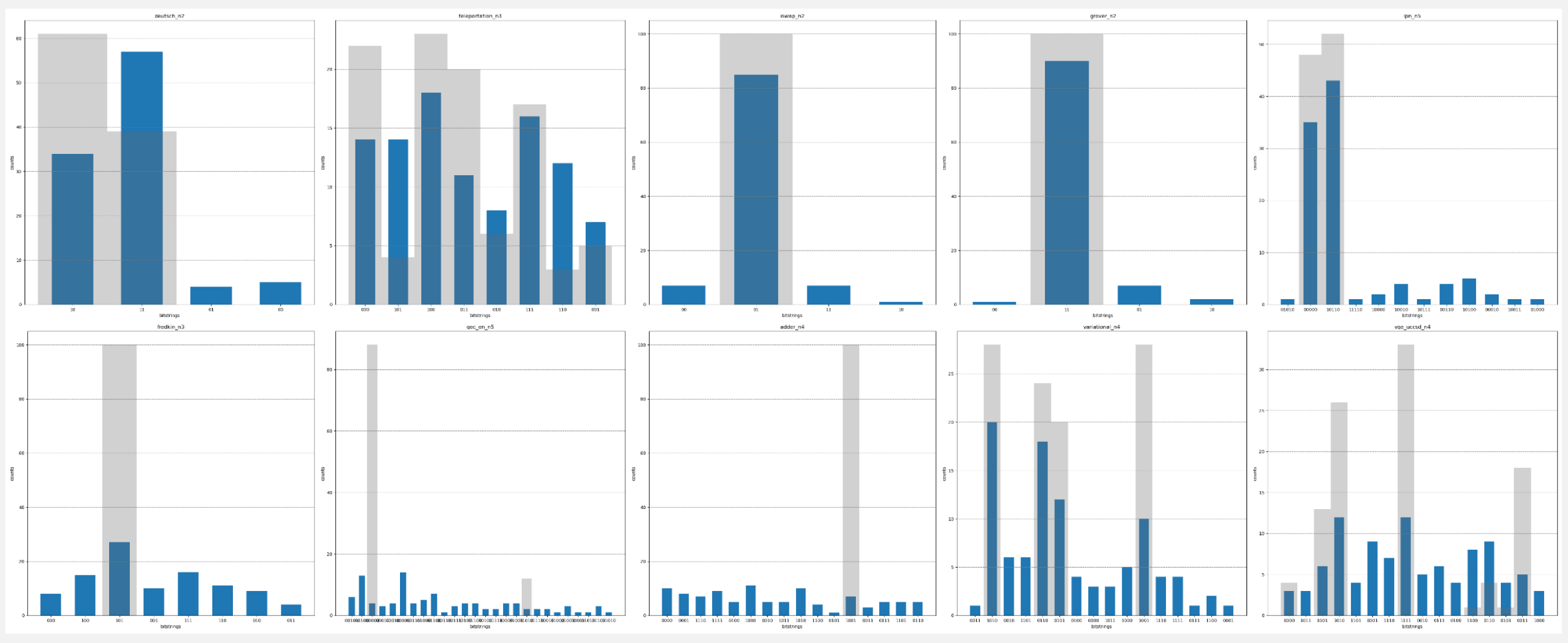
`Harmony` by IonQ, 11 qubit (ion-trap)



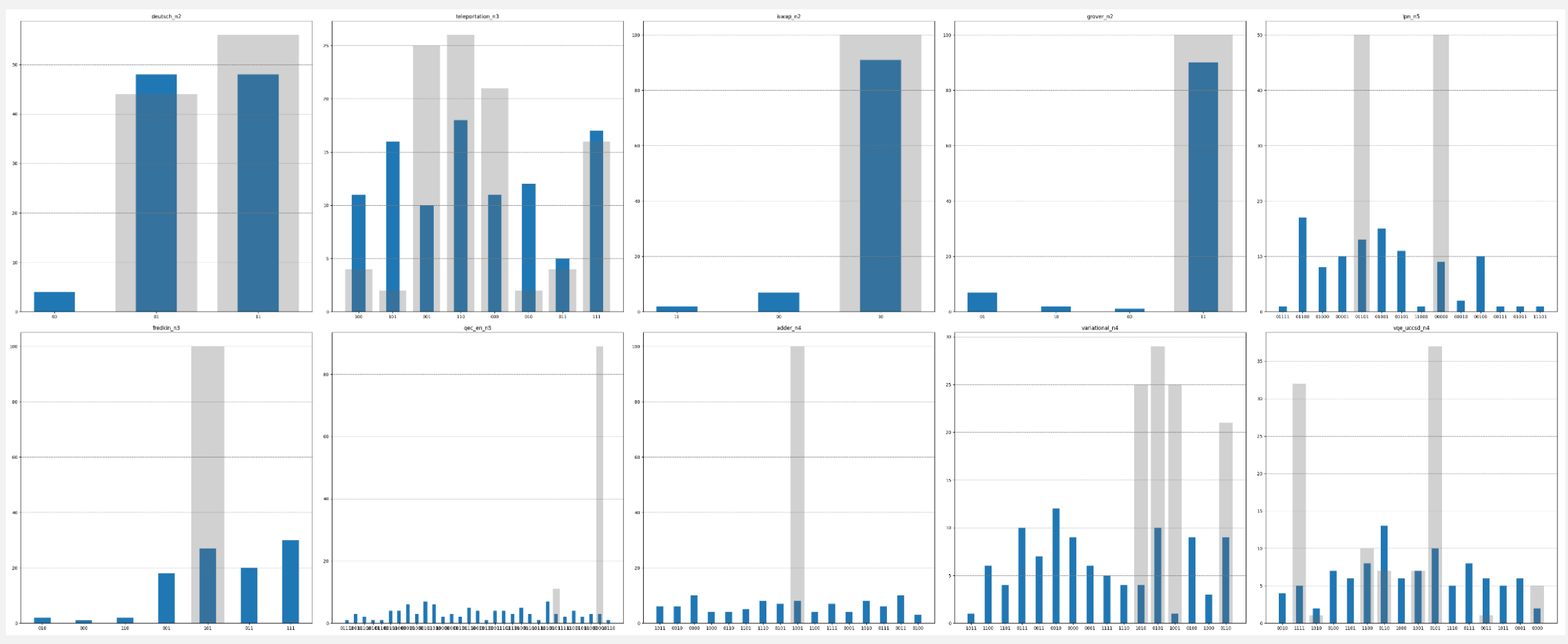
`Lucy` by Oxford Quantum Circuits, 8 qubit (superconducting)



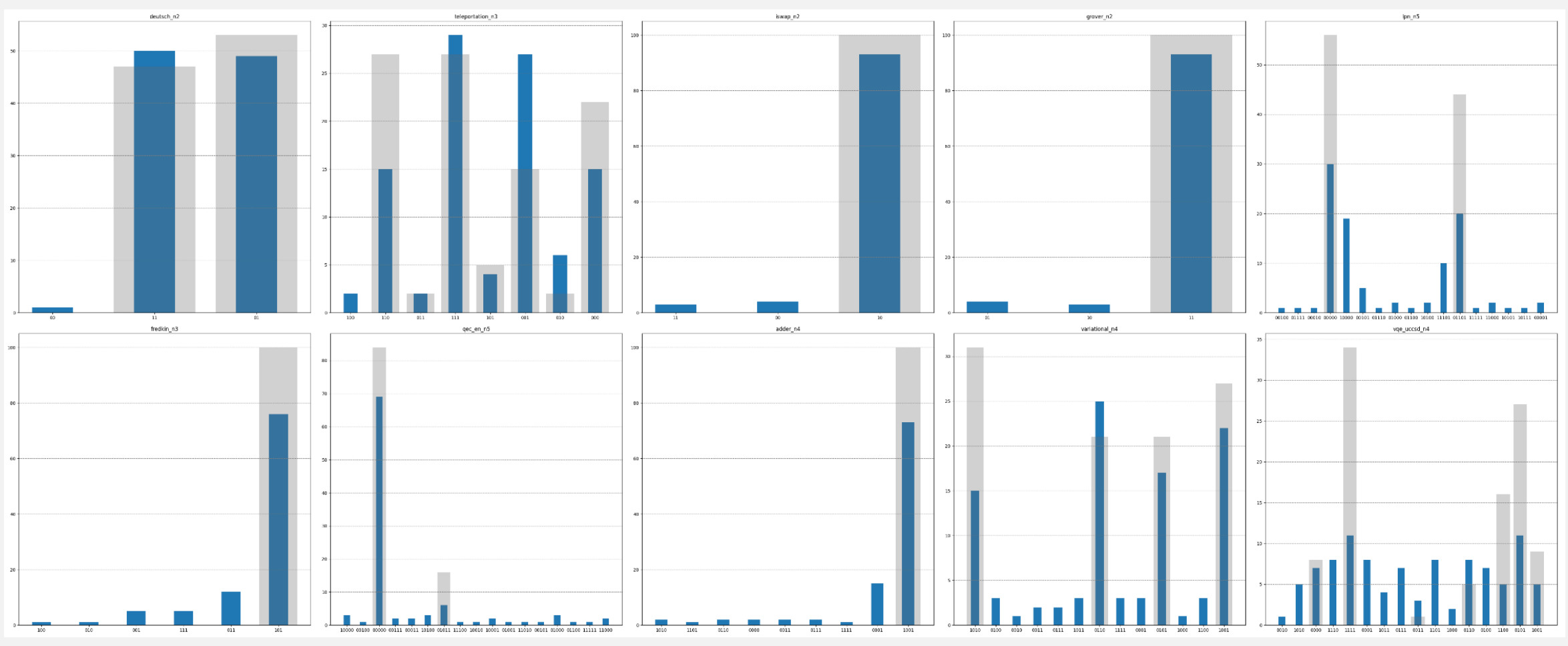
`Aspen-M-3` by Rigetti, 79 qubit (superconducting)



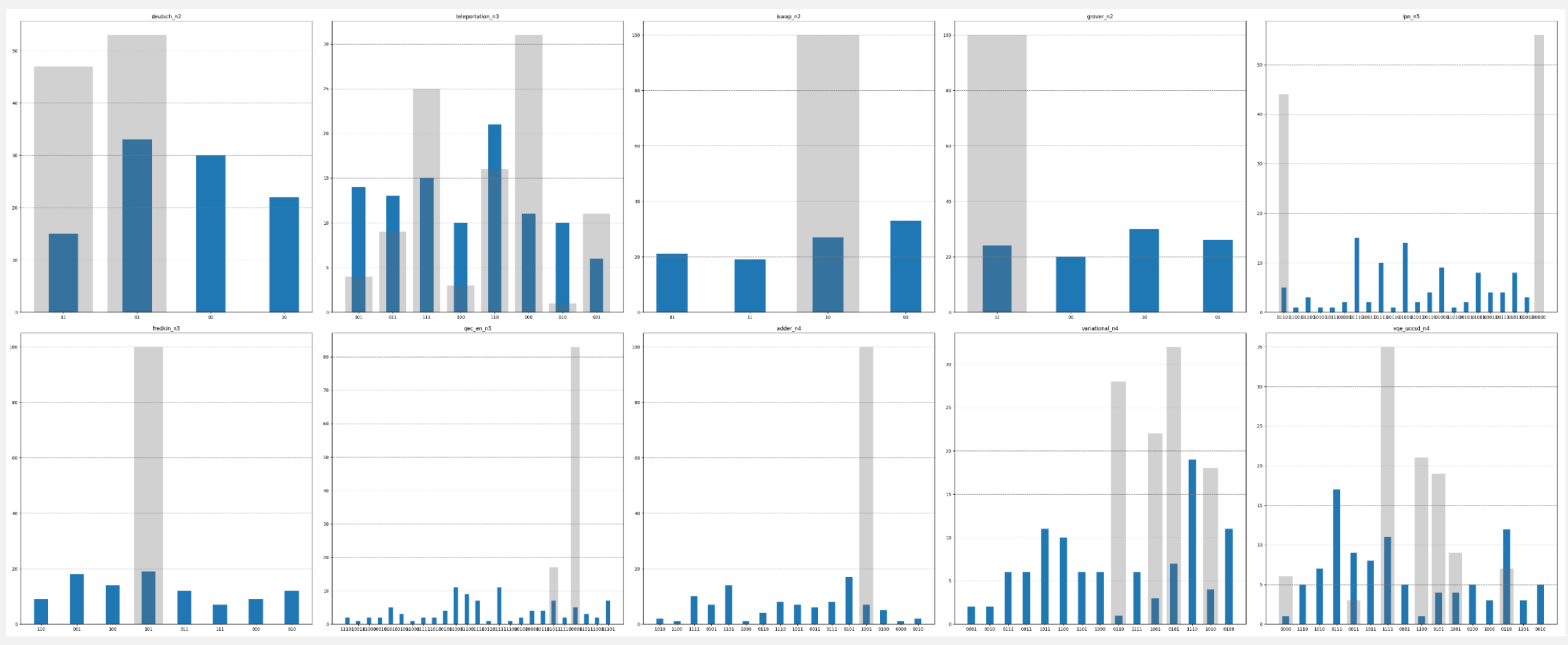
`ibm_nairobi by IBM, 7 qubit (superconducting)`



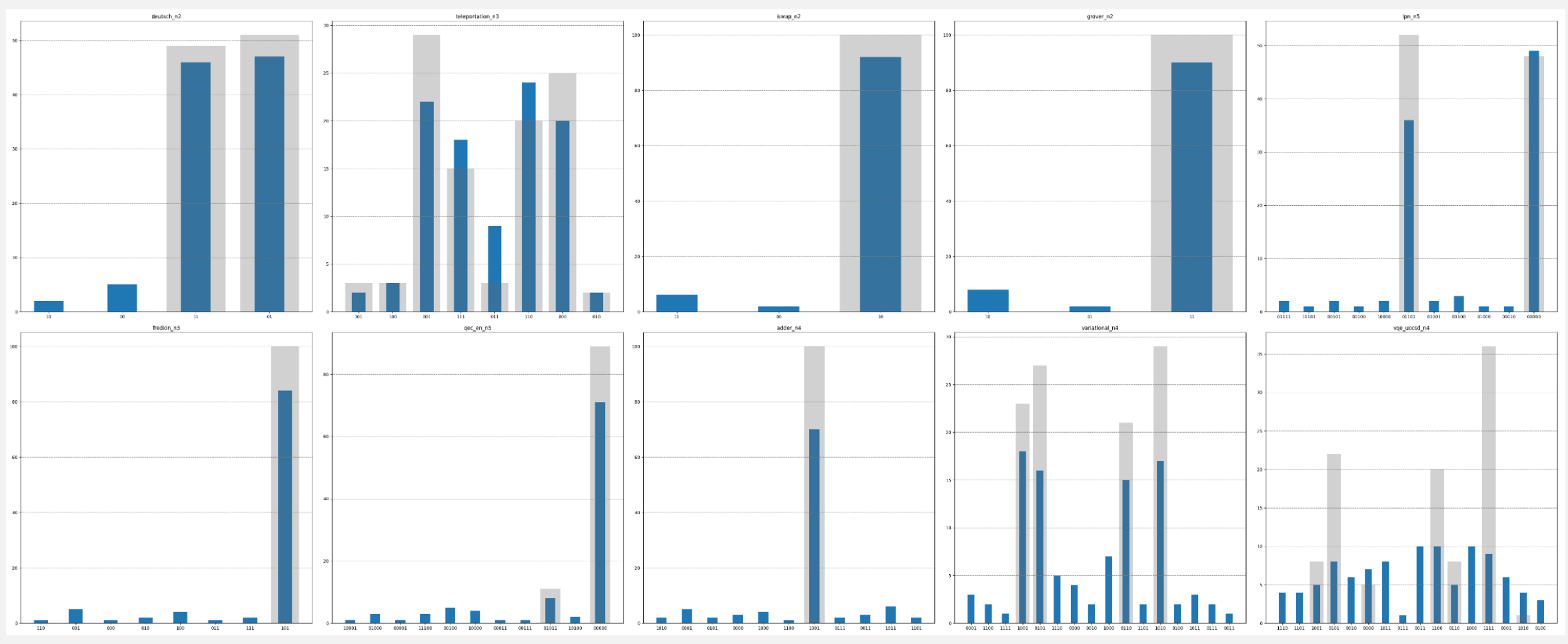
`ibm_oslo by IBM, 7 qubit (superconducting)`



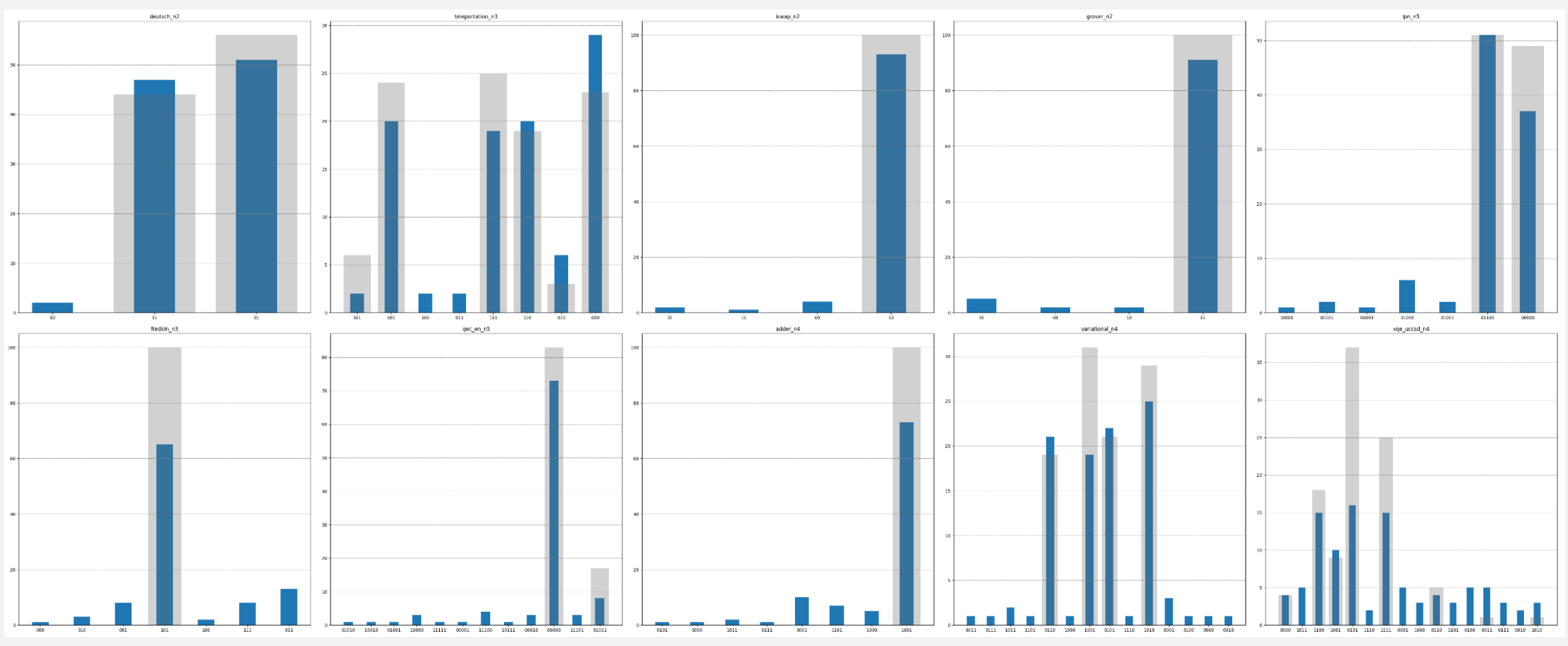
`ibmq_belem by IBM, 5 qubit (superconducting)`



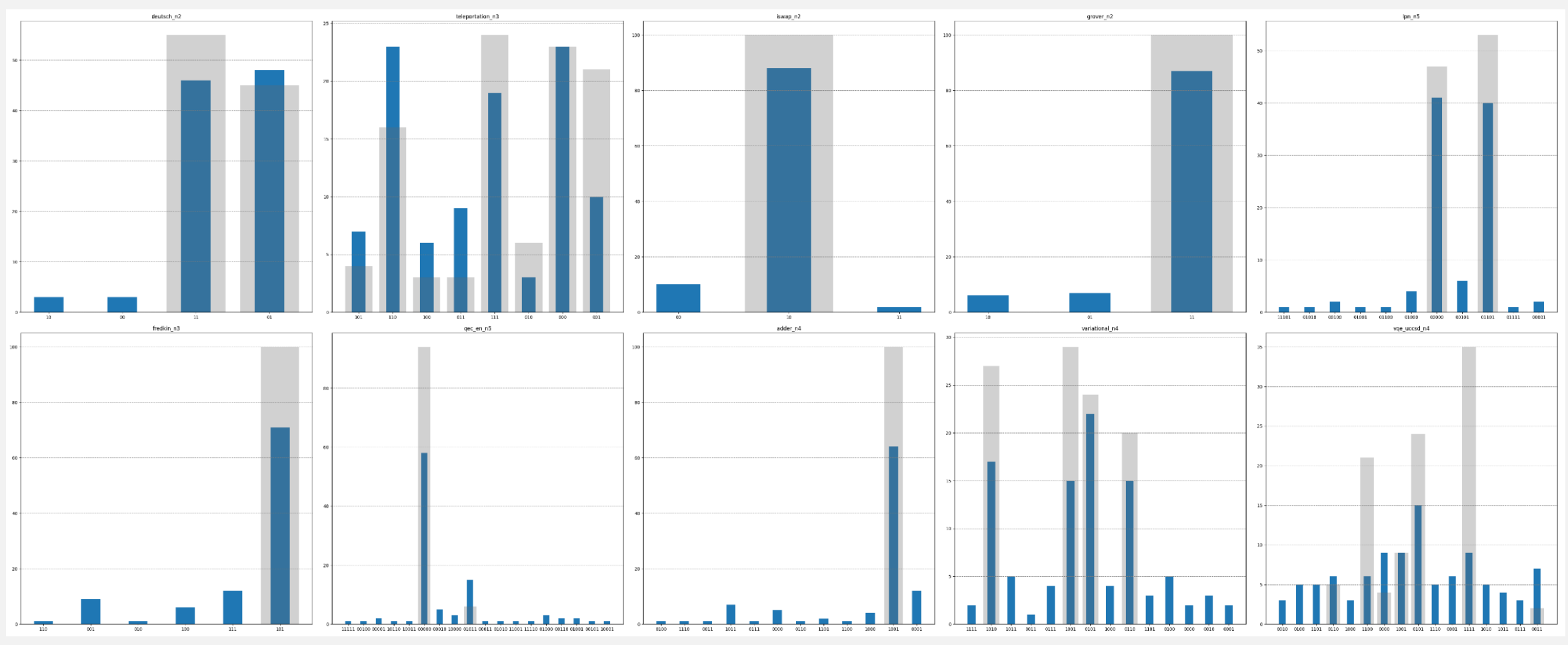
`ibmq_lima by IBM, 5 qubit (superconducting)`



`ibmq_manila by IBM, 5 qubit (superconducting)`



`ibmq_quito by IBM, 5 qubit (superconducting)`



DISCUSSION

- Current gate-model quantum processors can handle simple quantum computations with reasonably high fidelity.
- Due to the random nature of some quantum algorithms (e.g., Quantum teleportation), the performance might not be consistent across different quantum processors.
- Complex algorithms that use many CNOT connections, like the Variational algorithms, are still challenging for current quantum processors to compute.
- With active research on Quantum Error Correction and better hardware implementation technologies, the future of quantum computing is extremely promising.

REFERENCES

- Qiskit contributors, Qiskit: An Open-source Framework for Quantum Computing (2023). https://doi.org/10.5281/zenodo.7897504
- IBM Quantum (2021). https://quantum-computing.ibm.com/
- Amazon Web Services, Amazon Bracket (2020). https://aws.amazon.com/braket/
- A. Li, S. Stein, S. Krishnamoorthy, and J. Ang, Qasmbench: A low-level qasm benchmark suite for NISQ evaluation and simulation (2022).