Quantum Volume

Lev S. Bishop, Sergey Bravyi, Andrew Cross, Jay M. Gambetta, John Smolin

March 4, 2017

1 Executive Summary

As we build larger quantum computing devices capable of performing more complicated algorithms, it is important to quantify their power. The origin of a quantum computer's power is already subtle, and a quantum computer's performance depends on many factors that can make assessing its power challenging. These factors include:

1. The number of physical qubits;
2. The number of gates that can be applied before errors make the device behave essentially classically;
3. The connectivity of the device;
4. The number of operations that can be run in parallel.

Here we propose an architecture-neutral metric, the *quantum volume*, to summarize performance against these factors. The quantum volume measures the useful amount of quantum computing done by a device in space and time. Table 1 summarizes predicted quantum volumes for potential near-term devices.

Table 1: Quantum volume for some near-term devices

Device		
Topology	Error rate	Quantum Volume
IBM QX 5Q[a]	5×10^{-2}	16
5Q	10^{-2}	25
4×4	10^{-2}	36
4×4	10^{-3}	256
7×7	10^{-3}	256
7×7	10^{-4}	1296
10×10	10^{-4}	1296
10×10	10^{-5}	8100

[a] IBM Quantum Experience[1]

2 Introduction

As the community continues down the path toward more capable quantum devices, it is important to develop and apply metrics and tests that quantify capability. We propose architecture-neutral metrics to summarize the capability of short-depth quantum circuits in the non-fault-tolerant regime. It is challenging to compare devices with widely different performance characteristics. For example, a recent comparison[2] between the IBM Quantum Experience and an ion-based processor, pointed out that even with similar gate errors, the ion processor's greater connectivity allowed overall better performance for certain algorithms.

We emphasize that general metrics provide some overall sense of the quantum capabilities of a device for high-level comparisons, similar to the LINPACK benchmarks[3] for classical HPC. However, for a complete characterization and comparison of device capabilities against a specific task, there will be no alternative but to make a detailed investigation, including all relevant aspects and/or benchmarking the the target algorithms on the hardware in question.

Here we focus on hard-cutoff metrics such as 'can this device run a given algorithm?' rather than soft metrics such as 'how long will it take?' Thus, for example, we do not incorporate the gate speed, except indirectly as it affects the errors, nor do we give any credit for trivial parallelism by complete duplication of the quantum processor.

We propose an *effective error rate* ϵ_{eff}, specifying how well a device can implement arbitrary pairwise interactions between qubits. We further propose the *quantum volume* V_Q, combining the number of qubits n with the effective error rate in a difficult-to-game overall metric.

In Fig. 1, we show the requirements on n and on the 2-qubit gate error rate ϵ for a given V_Q, for differing qubit connectivities. This shows that the all-to-all connectivity has the least stringent error requirements, that the square lattice connectivity and the square lattice with diagonals have substantially tougher requirements, and a linear 1-d array of qubits is much tougher still.

3 Definitions of the metrics

3.1 Effective error rate

We want to abstract away many details, such as:

- the hardware-provided gate set;
- the qubit connectivity graph;
- varying fidelities of different operations;
- possibilities for circuit-rewriting and optimization;
- available paralellization of operations;
- etc.

We do this by specifying a *model algorithm*: performing a depth-1 circuit, constructed by random 2-qubit unitaries chosen uniformly over $SU(4)$ on a random pairing of the qubits. We then define ϵ_{eff} as the equivalent per-gate error rate that would lead to the same overall error rate. Thus, if the hardware supports $SU(4)$ operations directly, has an all-to-all connectivity with unlimited gate paralellism, all gates having identical error rate ϵ, then $\epsilon_{\text{eff}} = \epsilon$.

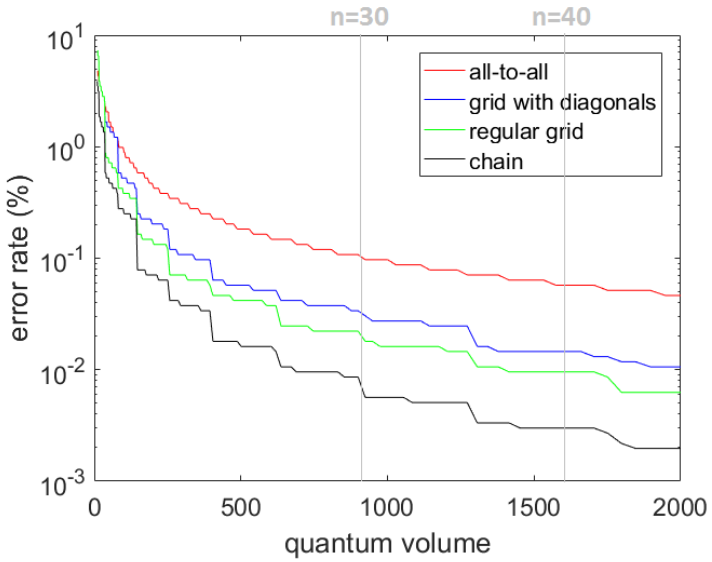

Figure 1: The required two-qubit gate error rate, ϵ, to acheive a given quantum volume, V_Q, in the limit where there are more than enough qubits available. Vertical lines indicate maximum acheivable volume n^2 for given number of qubits.

If the connectivity is limited, then it will be necessary to insert SWAP gates to permute the qubits in order to allow the necessary gates. (It is not required to permute the qubits back to their initial order).

If there is only a finite set of available gates, then it will be necessary to approximate the desired unitaries in the manner of Solovay-Kitaev, trading off approximation error against the added noise from long gate sequences.

For a system with only local connectivity but the ability to perform fast measurement and feedback, it may prove preferable to use teleportation to couple distant parts of the device rather than long chains of SWAP gates to directly implement the permutations.

Other special features and limitations of the hardware must be dealt with in similar manner. Note that ϵ_{eff} depends not only on the gate error rates and connectivity, but also on the sophistication of the scheduling algorithm responsible for mapping the model algorithm to the hardware. We permit the scheduling to occur off-line. Both hardware and software improvements will thus impact ϵ_{eff}.

The rationale for chosing the model algorithm to be built from pairwise interactions rather than sampling over the full $SU(2^n)$ space of unitaries on the system, is that the space of all unitaries, while it can always be constructed from sequences of pairwise interactions, is exponentially large and would include, for example, unitaries that cannot be synthesized as polynomial-sized quantum circuits. There may be specialized algorithms that do not require long-range interactions but can be mapped directly to the connectivity of the underlying device (for example physics calculations in in 2 dimensions may map nicely to a square lattice of qubits), but random pairings are representative for general-purpose short-depth algorithms.

3.2 Quantum volume

For any given instance of a quantum algorithm, there is a lower bound on the number of qubits, n, required to run the algorithm as well as the achievable circuit depth, $d \simeq 1/(n\epsilon_{\text{eff}})$ needed to execute the algorithm with reasonable fidelity to the correct answer.

If it is desired to have a single metric for comparing systems, then it seems reasonable to take the product $dn = 1/\epsilon_{\text{eff}}$. However, this has some undesirable properties in that it can be gamed in various ways. For example, in many cases the best ϵ_{eff} will result from very few qubits, even $n = 2$, since in this case there will be less connectivity and paralellization overhead, and fewer issues with crosstalk between qubits. But clearly $n = 2$ is a completely uninteresting limit, where all algorithms can be trivially simulated classically. Therefore, we define $V_Q = \min(n, d)^2$, and since ϵ_{eff} and d in general depend on n, we should maximize over the number of active qubits, n', choosing a subset of n on which to execute the model algorithm (the remaining qubits may nevertheless participate as helpers, for example to reduce the permutations needed to implement the model algorithm)

$$V_Q = \max_{n' \leq n} \min\left[n', \frac{1}{n'\epsilon_{\text{eff}}(n')}\right]^2. \tag{1}$$

This metric quantifies the space-time volume occupied by a model circuit with random two-qubit gates that can be reliably executed on a given device.

4 Estimation of metrics for different connectivities

We propose an heuristic algorithm for finding a good set of SWAPs to permute qubits in order to implement an instance of the model algorithm for a given qubit connectivity. This provides a lower-bound on on ϵ_{eff} and V_Q that we expect should be fairly tight in most cases.

Specifically, we perform a greedy search for the lowest-depth circuit to permute the qubits such that all qubit pairs (u_j, v_j) that interact for this instance are brought into adjacency. At each step we implement the SWAPs that reduce the total distance $\sum_j D(u_j, v_j)$ between these qubits, for some distance function D

$$D(u, v) = (1 + |\xi_{u,v}|)D_0(u, v)^2, \tag{2}$$

where D_0 is the distance between u and v for a given connectivity graph, and $\xi_{u,v}$ are random variables drawn from $N(0, 1/n)$. Since the algorithm is randomized, we repeat for many realizations of $\xi_{u,v}$ and choose the lowest computed depth, r.

Averaging r over many instances of the model algorithm gives the effective error rate as $\epsilon_{\text{eff}} = \epsilon(\bar{r} + 1)$, where we assume that all SWAP gates and the needed $SU(4)$ interactions all can be done with constant error ϵ.

The square lattice numbers in Fig. 1 somewhat underestimate the overhead $\bar{r}$ for the Boixo proposal [4] due to in that case there is restricted parallelism available, which we expect to give an additional factor of approximately 2. The grid with diagonals calculation somewhat underestimates the overhead for the IBM skew-square connectivity because it assumes all diagonals are available whereas in current proposals diagonals are only available on odd plaquettes [5].

References

[1] The IBM quantum experience. URL http://www.research.ibm.com/quantum/.

[2] N. M. Linke, D. Maslov, M. Roetteler, S. Debnath, C. Figgatt, K. A. Landsman, K. Wright, and C. Monroe. Experimental Comparison of Two Quantum Computing Architectures. *arXiv:1702.01852 [quant-ph]*, February 2017. URL http://arxiv.org/abs/1702.01852. arXiv: 1702.01852.

[3] The LINPACK benchmark. URL https://www.top500.org/project/linpack/.

[4] Sergio Boixo, Sergei V. Isakov, Vadim N. Smelyanskiy, Ryan Babbush, Nan Ding, Zhang Jiang, John M. Martinis, and Hartmut Neven. Characterizing Quantum Supremacy in Near-Term Devices. *arXiv:1608.00263 [quant-ph]*, July 2016. URL http://arxiv.org/abs/1608.00263. arXiv: 1608.00263.

[5] Jay M. Gambetta, Jerry M. Chow, and Matthias Steffen. Building logical qubits in a superconducting quantum computing system. *npj Quantum Information*, 3(1):2, January 2017. ISSN 2056-6387. doi: 10.1038/s41534-016-0004-0. URL http://www.nature.com/articles/s41534-016-0004-0.