Computing output weights of constant-depth, geometrically-local quantum circuits with polynomial additive error.

Nolan J. Coble[*,†], Matthew Coudron[‡]

November 21, 2020

In this work we present a classical algorithm that, for any geometrically-local, constant-depth quantum circuit C, and any bit string $x \in \{0,1\}^n$, can compute the quantity $|\langle 0^{\otimes n}| C |x\rangle|^2$ to within any inverse-polynomial additive error in quasi-polynomial time. See Theorem 1 below for the full statement. It is known that it is $\#P$-hard to compute this same quantity to within 2^{-n^2} additive error [Mov20]. The previous best known algorithm for this problem used $O(2^{n^{1/3}}\mathsf{poly}(1/\epsilon))$ time to compute probabilities to within additive error ϵ [BGM20]. Notably, the [BGM20] paper included an elegant polynomial time algorithm for estimating the quantity $|\langle 0^{\otimes n}| C |x\rangle|^2$ when C is a 2D, geometrically-local, constant-depth circuit. That algorithm makes a novel use of 1D Matrix Product States carefully tailored to the 2D geometry of the circuit in question. Surprisingly, it is not clear that it is possible to extend this use of MPS to address the case of 3D circuits in polynomial time. This raises a natural question as to whether the computational complexity of the 3D problem might be drastically higher than that of the 2D problem. In this work we address this question by exhibiting a quasi-polynomial time algorithm for the 3D case. We believe that our algorithm extends naturally to any fixed dimension D by induction on the dimension, but we focus on the 3D case, as the simplest unresolved case, for concreteness. Furthermore, we show that, under a natural, polynomial-time-checkable condition on the circuit C, our algorithm runs in polynomial time. This highlights the possibility that the super-polynomial time cost of our algorithm in the worst-case might be due to limitations in our analysis. In order to surpass the technical barriers encountered by previously known techniques we are forced to pursue a novel approach: Constructing a recursive sub-division of the given 3D circuit using carefully designed block-encodings. See the Results and Techniques section below for more.

Our algorithm has a Divide-and-Conquer structure, demonstrating how to approximate the desired quantity via several instantiations of the same problem type, each involving 3D-local circuits on at most half the number of qubits as the original. This division step is then applied recursively, expressing the original quantity as a weighted sum of smaller and smaller 3D-local quantum circuits. A central technical challenge is to control correlations arising from the entanglement that may exist between the different circuit "pieces" produced this way. We believe that the division step, which makes a novel use of block-encodings [GSLW19], together with an Inclusion-Exclusion style argument to reduce error, may be of interest for future research on constant-depth quantum circuits.

Background Our motivation for estimating the quantity $|\langle 0^{\otimes n}| C |x\rangle|^2$ lies in the relationship between this computational problem and a popular paradigm for near-term quantum computing experiments. Many schemes for obtaining a quantum computational advantage with near-term quantum hardware are motivated by mathematical results proving the computational hardness of sampling from near-term quantum circuits. Indeed, an extensive line of research [AA11, BJS11, BMS17, NSC+17, BFNV19, AA11] has established hardness of sampling results for a wide variety of quantum circuit classes, including the

[*]Authors are listed alphabetically.
[†]`ncoble@terpmail.umd.edu`
[‡]`mcoudron@umd.edu` - Corresponding Author

constant-depth, geometrically-local quantum circuits that are the focus of this work. It has even been shown, under several computational assumptions, that there is no classical polynomial time algorithm which, given a constant-depth, geometrically-local quantum circuit, K, can produce samples whose distribution lies within a constant, in the ℓ_1 distance, of the output distribution of K in the computational basis [BVHS+18]. Note that sampling from the output state of a quantum circuit is not the same task as estimating the weights of the outputs, especially if the latter estimate has inverse polynomial additive error. However, we believe that understanding computational problems of the latter type will help us determine whether constant-depth quantum circuits can produce a quantum advantage for a Decision problem. Our result represents only one step in this direction, as we do not cover the case of constant-depth quantum circuits which are not geometrically-local, and the computational power of constant-depth quantum circuits with classical pre-processing and post-processing remains uncharacterized. Our result resolves a natural question which is prerequisite to answering these others.

A number of classical simulation algorithms for various near-term quantum architectures have already been proposed [DHKLP20, HZN+20, NPD+20]. These results highlight the subtle nature of identifying a quantum advantage based on these architectures, even in the case of sampling problems. They also contrast with the computational problem studied in Theorem 5 in [BGM20]: The problem of estimating output probabilities of 2D-local constant depth circuits to inverse polynomial additive error in polynomial time. In fact, the original algorithm in [BGM20], actually estimates quantities of the form $\langle 0^{\otimes n}| C^{\dagger} \left(\otimes_{i=1}^{n} P_i\right) C |0^{\otimes n}\rangle$, where each $P_i \in \{X, Y, Z, I\}$ is a single-qubit Pauli observable operator. However, it is straightforward to convert their algorithm to compute the quantity $\langle 0^{\otimes n}| C^{\dagger} \left(\otimes_{i=1}^{n} |x_i\rangle \langle x_i|\right) C |0^{\otimes n}\rangle = |\langle 0^{\otimes n}| C |x\rangle|^2$, $x \in \{0,1\}^n$, instead. This focus of [BGM20] raises a pertinent observation. While it is hard to sample from constant-depth quantum circuits, it is still unresolved whether it is hard to estimate any property of such a circuit which could have been computed using a polynomial number of samples from the output of the quantum circuit itself. In particular: A polynomial number of samples from a 2D-local, constant-depth quantum circuit only allows one to estimate output probabilites of that circuit to inverse polynomial additive error. But, it is shown in Theorem 5 of [BGM20] that this same task can be done in classical polynomial time! Is there a well-defined Decision problem which can be solved using only a polynomial number of samples from such a quantum circuit, together with classical post-processing, and yet cannot also be efficiently solved using classical computing alone? This is unknown.

Techniques and Results: The algorithm for 2D circuits presented in Theorem 5 of [BGM20] makes a novel use of 1D Matrix Product States, carefully tailored to the 2D geometry of the circuit in question. However, the authors of [BGM20] point out that it is not clear that it is possible to extend this use of MPS to address the case of 3D circuits in polynomial time. Instead they provide a sub-exponential time algorithm for the 3D case, which has time complexity $O(2^{n^{1/3}} \mathrm{poly}(1/\epsilon))$ for computing the desired quantity to within additive error ϵ. In other words, their algorithm is exponential in the length of one side of the 3D cubic lattice of qubits on which the 3D circuit acts. In this work we introduce a new set of techniques culminating in a divide-and-conquer algorithm which moves beyond this barrier, and solves the 3D case in quasi-polynomial time.

Our algorithm has a divide-and-conquer structure with the goal being to divide the circuit C into pieces, and reduce the original problem to a small number of new 3D-circuit problems involving circuits on only a fraction of the number of qubits as the original. This division step requires the ability to construct Schmidt vectors of the state $C|0^{\otimes n}\rangle$, across a given cut, via a constant-depth geometrically local quantum circuit, so that the new subproblems can be expressed as smaller instantiations of the original problem type. We accomplish this through the use of block-encodings, a technique designed for quantum algorithms [GSLW19], but used here as a subroutine of a classical simulation algorithm instead. However, to date, we are only able to construct, as a block-encoding circuit, the *leading* Schmidt vector across certain "heavy" cuts. Due to this restriction we are forced to use a novel division step in our Divide-and-Conquer approach. Instead of dividing about a single cut and constructing many of its Schmidt vectors as constant-depth geometrically local block-encodings, we must divide across many cuts and construct only their leading Schmidt vectors. Interestingly, this process can still lead to low approximation error via an Inclusion-Exclusion style argument.

These techniques culminate in a worst-case quasi-polynomial time algorithm for 3D circuits, which is our main result:

Theorem 1. *Let C be any depth-d, 3D geometrically local quantum circuit on n qubits. Algorithm 1, $\mathcal{A}_{full}(C, \mathcal{B}, \delta)$ (where $\mathcal{B}$ is the base case algorithm, chosen to be the algorithm in Theorem 5 of [BGM20]) will produce the scalar quantity $|\langle 0^{\otimes n}| C |0^{\otimes n}\rangle|^2$ to within δ error in time*

$$T(n) = 2^{\mathsf{polylog}(n)(1/\delta)^{1/\log^2(n)}} \cdot 2^{d^3} \tag{1}$$

(See technical abstract for the precise definition of Algorithm 1, $\mathcal{A}_{full}(C, \mathcal{B}, \delta)$.)

Note that, for any $\delta = \Omega(1/n^{\log(n)})$, we have $(1/\delta)^{1/\log^2(n)} = O(1)$, and this runtime is quasi-polynomial. In particular, for any $\delta(n)$ which scales inverse-polynomially (or even for some inverse-quasi-polynomial scaling), the algorithm runs in quasi-polynomial time. Furthermore, under a natural, polynomial-time-checkable assumption on the circuit C, we obtain a polynomial time algorithm:

Theorem 2. *Let C be any depth-d, 3D geometrically local quantum circuit on n qubits. If we assume Assumption 33, then Algorithm 3, $\mathcal{A}_{const}(C, \mathcal{B}, \delta)$ (where $\mathcal{B}$ is the base case algorithm chosen to be as in Theorem 5 of [BGM20]), will approximate the scalar quantity $|\langle 0^{\otimes n}| C |0^{\otimes n}\rangle|^2$ to within δ additive error in time*

$$T(n) = \mathsf{poly}(n, 2^{(1/\delta)^{1/\log^2(n)}}) \cdot 2^{d^3}. \tag{2}$$

(See technical abstract for the precise definition of Algorithm 3, $\mathcal{A}_{const}(C, \mathcal{B}, \delta)$.)

Note that, for any $\delta = \Omega(1/(n)^{\log(n)})$, we have $(1/\delta)^{1/\log^2(n)} = O(1)$, and this runtime is polynomial.

References

[AA11] Scott Aaronson and Alex Arkhipov. The computational complexity of linear optics. In *Proceedings of the Forty-Third Annual ACM Symposium on Theory of Computing*, STOC '11, page 333–342, New York, NY, USA, 2011. Association for Computing Machinery. URL: https://doi.org/10.1145/1993636.1993682, doi:10.1145/1993636.1993682.

[AAB+19] Frank Arute, Kunal Arya, Ryan Babbush, Dave Bacon, Joseph C. Bardin, Rami Barends, Rupak Biswas, Sergio Boixo, Fernando G. S. L. Brandao, David A. Buell, and et al. Quantum supremacy using a programmable superconducting processor. *Nature*, 574(7779):505–510, Oct 2019. doi:10.1038/s41586-019-1666-5.

[Aar14] Scott Aaronson. The equivalence of sampling and searching. *Theory of Computing Systems*, 55(2):281–298, Aug 2014. doi:10.1007/s00224-013-9527-3.

[BFNV19] Adam Bouland, Bill Fefferman, Chinmay Nirkhe, and Umesh Vazirani. On the complexity and verification of quantum random circuit sampling. *Nature Physics*, 15(2):159–163, Feb 2019. doi:10.1038/s41567-018-0318-2.

[BG16] Sergey Bravyi and David Gosset. Improved classical simulation of quantum circuits dominated by clifford gates. *Phys. Rev. Lett.*, 116:250501, Jun 2016. URL: https://link.aps.org/doi/10.1103/PhysRevLett.116.250501, doi:10.1103/PhysRevLett.116.250501.

[BGK18] Sergey Bravyi, David Gosset, and Robert König. Quantum advantage with shallow circuits. *Science*, 362(6412):308, Oct 2018. doi:10.1126/science.aar3106.

[BGKT20] Sergey Bravyi, David Gosset, Robert König, and Marco Tomamichel. Quantum advantage with noisy shallow circuits. *Nature Physics*, 16(10):1040–1045, Oct 2020. doi:10.1038/s41567-020-0948-z.

[BGM20] Sergy Bravyi, David Gosset, and Ramis Movassagh. Classical algorithms for quantum mean values. QIP, 2020. URL: https://arxiv.org/abs/1909.11485.

[BIS+18] Sergio Boixo, Sergei V. Isakov, Vadim N. Smelyanskiy, Ryan Babbush, Nan Ding, Zhang Jiang, Michael J. Bremner, John M. Martinis, and Hartmut Neven. Characterizing quantum supremacy in near-term devices. *Nature Physics*, 14(6):595–600, Jun 2018. doi:10.1038/s41567-018-0124-x.

[BJS11] Michael J. Bremner, Richard Jozsa, and Dan J. Shepherd. Classical simulation of commuting quantum computations implies collapse of the polynomial hierarchy. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 467(2126):459–472, Feb 2011. doi:10.1098/rspa.2010.0301.

[BMS17] Michael J. Bremner, Ashley Montanaro, and Dan J. Shepherd. Achieving quantum supremacy with sparse and noisy commuting quantum computations. *Quantum*, 1:8, Apr 2017. URL: http://dx.doi.org/10.22331/q-2017-04-25-8, doi:10.22331/q-2017-04-25-8.

[BVHS+18] Juan Bermejo-Vega, Dominik Hangleiter, Martin Schwarz, Robert Raussendorf, and Jens Eisert. Architectures for quantum simulation showing a quantum speedup. *Phys. Rev. X*, 8:021010, Apr 2018. URL: https://link.aps.org/doi/10.1103/PhysRevX.8.021010, doi:10.1103/PhysRevX.8.021010.

[DHKLP20] Alexander M. Dalzell, Aram W. Harrow, Dax Enshan Koh, and Rolando L. La Placa. How many qubits are needed for quantum computational supremacy? *Quantum*, 4:264, May 2020. URL: http://dx.doi.org/10.22331/q-2020-05-11-264, doi:10.22331/q-2020-05-11-264.

[GSLW19] András Gilyén, Yuan Su, Guang Hao Low, and Nathan Wiebe. Quantum singular value transformation and beyond: exponential improvements for quantum matrix arithmetics. STOC, 2019. URL: https://arxiv.org/pdf/1806.01838.pdf.

[HZN+20] Cupjin Huang, Fang Zhang, Michael Newman, Junjie Cai, Xun Gao, Zhengxiong Tian, Junyin Wu, Haihong Xu, Huanjun Yu, Bo Yuan, Mario Szegedy, Yaoyun Shi, and Jianxin Chen. Classical simulation of quantum supremacy circuits, 2020. arXiv:2005.06787.

[Mov20] Ramis Movassagh. Quantum supremacy and random circuits. QIP, 2020. URL: https://arxiv.org/pdf/1909.06210.pdf.

[NPD+20] John Napp, Rolando L. La Placa, Alexander M. Dalzell, Fernando G. S. L. Brandao, and Aram W. Harrow. Efficient classical simulation of random shallow 2d quantum circuits, 2020. arXiv:2001.00021.

[NSC+17] Alex Neville, Chris Sparrow, Raphaël Clifford, Eric Johnston, Patrick M. Birchall, Ashley Montanaro, and Anthony Laing. Classical boson sampling algorithms with superior performance to near-term experiments. *Nature Physics*, 13(12):1153–1157, Dec 2017. doi:10.1038/nphys4270.