Quantum Resource Estimation of Arithmetic Primitives

Ethan Hansen
Department of Physics
University of Washington
Seattle, USA
ethanrh@uw.edu

Sanskriti Joshi
Department of Electrical and Computer Engineering
University of Washington
Seattle, USA
sjoshi3@uw.edu

Hannah Rarick
Department of Physics
University of Washington
Seattle, USA
rarichan@uw.edu

2023 IEEE International Conference on Quantum Computing and Engineering (QCE) | 979-8-3503-4323-6/23/$31.00 ©2023 IEEE | DOI: 10.1109/QCE57702.2023.10290

I. Introduction

A major component of RSA cryptanalsysis, factoring a number that is a product of two large primes is computationally intensive. Using Shor's algorithm, there is a possibility of the factorization taking a fraction of the time on a quantum computer. To implement this level of algorithm, an analysis of the resources required would help with understanding the requirements for the hardware based on the software implementation and vice versa.

To date, researchers are investigating transforming classical algorithms for arithmetic, like multiplication, into quantum algorithms to be used on quantum processors [1]. Although there are publications related to the various quantum algorithms for multiplication, none of them provide an analysis of the resource estimation using Microsoft Azure Quantum Resource Estimator [2] - [3]. Understanding the resources required to run an algorithm helps define a scale for how large and resilient to noise a quantum computer has to be to effectively run the algorithm [5].

II. Multiplication Algorithms

This project will focus on resource estimation for multiplication algorithms. Two different classical algorithms were used in addition to schoolbook multiplication: Karatsuba multiplication and windowed multiplication. One barrier to directly implementing classical algorithms to quantum algorithms is that not all classical algorithms are reversible. This irreversibly introduces decoherence in the circuit, equating to a noisier algorithm. We will note how the decoherence is addressed in each of the quantum multiplication algorithms.

Schoolbook multiplication is the method taught in grade school for long multiplication. This multiplication requires $\mathcal{O}(n^2)$ operations.

A. Karatsuba multiplication

Karatsuba multiplication requires a sub-quadratic number of operations by recursively multiplying, while using those results to calculate the final answer. For example, to perform the multiplication of integers a and b, each integer gets broken up such that $a = c + 2^h d$ and $b = e + 2^h f$. Using the results of recursively multiplying $ce, df, (c + d)(e + f)$ yields the desired result [2]. In the quantum implementation of the algorithm, intermediate multiplication values are added directly to sections of the output register to remove decoherence [2]. Notably, this method of removing decoherence requires the same $\mathcal{O}(n)$ space usage and $\mathcal{O}(n^{\lg 3})$ operation count as the original classical algorithm.

B. Windowed multiplication

Windowed arithmetic reduces the number of operations counts by using classically precomputed look-up tables to merge together operations. While this makes the number of operations for the algorithm much smaller, generating the look-up table is more resource dense than the classical multiplications [3]. Additionally, measurement based uncomputations of the look-up tables can be performed to remove decoherence [3]. This method has asymptotic Toffoli gate count of $\mathcal{O}(\frac{n^2}{\lg n})$.

III. Methods

We will be working with Microsoft's quantum resource estimation tool to investigate and compare the resources required for different integer multiplication algorithms. The main estimation tool that we will utilize is the Azure Quantum Resource Estimator, referred throughout as the cloud resource estimator. The Azure Quantum Resource Estimator is run on Microsoft's machines and is accessed by uploading a quantum circuit (constructed in Q#) to the cloud via the Azure Portal. The cloud resource estimator estimates the resources (physical and logical) to implement a quantum circuit in a fault tolerant manner. This method takes into account the one- and two-qubit gate times and error rates of a physical quantum processor, as well as the quantum error correction scheme being utilized to form logical qubits. The cloud resource estimator uses provided presets for these gate times and error rates based on three different qubit platforms: gate-based superconducting, trapped-ion systems and Majorana systems, which have yet to be implemented. There are two presets for each platform, one realistic estimate and one optimistic estimate for the future gate times and error rates of a fault-tolerant quantum processor based upon the qubit platform. In Sec. IV-A, we will be utilizing the default qubit platform (gate-based quantum processor with 50 ns gate times, 100 ns measurement times, and 10^{-3} single- and two-qubit gate errors) and the default

979-8-3503-4323-6/23/$31.00 ©2023 IEEE
DOI 10.1109/QCE57702.2023.10290

Stabilizer error correction scheme to investigate the physical-space complexity and the time complexity. In Sec. IV-B, we will investigate estimated physical qubits and algorithm runtime for performing plus-equal multiplication of 2048-bit integers across the six platform presets provided by the resource estimator. For the Majorana platform, we will utilize floquet quantum error correction scheme as it provides an advantage over the surface code. For the other platforms, we will utilize the surface code.

To date, literature on resource estimation for multiplication algorithms uses the QCTraceSimulator implemented in Microsoft's Quantum Development Kit (QDK) for Q#. This tool estimates the resources of quantum algorithms at the logical level and is agnostic of any physical constraints related to implementing the algorithm on quantum hardware. In this project, we will be refactoring these algorithms utilizing the newer Microsoft resource estimation tool (Azure Quantum Resource Estimator), which estimates resources at the physical level and accounts for quantum error correction schemes, qubit gate and measurement times, and hardware implementation. Taking advantage of new metrics allotted by the cloud resource estimator (algorithm runtime and required physical qubits), we will make use of the resource estimator to investigate whether any of the multiplication algorithms can provide a significant advantage over the schoolbook algorithm.

IV. Results

All of the results show in this paper will be from cloud estimator rather than the QCTraceSimulator, as trace simulator results are already given in [2].

A. Physical-Space, Time, and T-State Complexity

We analyze the physical-space complexity of the Karatsuba, schoolbook, and windowed algorithms in the left plot of Fig. 4, where the vertical axis is the physical qubits (in millions) divided by the bit-size of the inputs, n. We observe that schoolbook multiplication utilizes the least amount of physical qubits for all factor sizes plotted, however windowed closely follows the same trend as schoolbook. We also see where karatsuba kicks in for 32-bit integers, but yields no advantage in terms of space.

The T-state complexity, shown in the right plot of Fig 4, follows a similar trend as the time complexity. This correlated behaviour is to be expected given the much larger time-cost of producing high fidelity T-gates/T-states compared to Clifford gates. The T-state complexity scales as $\mathcal{O}\left(n^2\right)$, while the time complexity has some non-trivial scaling for lower bit-sizes, but appears to converge to $\mathcal{O}\left(n^2\right)$ scaling for the larger bit-sizes shown.

B. Estimated Resources Across Qubit Platforms

We now analyze how the resources for each algorithm compared across different platforms, investigating the three qubit platforms currently provided by the Azure Quantum Resource Estimator: superconducting, trapped ion, and Majorana systems. For each system, we consider a realistic platform with realistic gate/measurement times and errors and a hopeful platform with optimistic realistic gate/measurement times and errors (see [4] for more on custom resource estimation parameters). We show the algorithm runtime and required physical qubits for each algorithm, using a factor size of 2048 across the six different platforms, in Fig. 3.

As expected from Fig. 3, we see that the windowed algorithm utilizes slightly more physical qubits than schoolbook; however, the algorithm runtime is shorter for windowed. The difference between windowed and schoolbook is minimal on the optimistic Majorana system: 5 million qubits for windowed compared to 4 million qubits for schoolbook on the optimistic Majorana system. This difference is even more negligible for the realistic superconducting platform, 26 million qubits for windowed compared to 24.2 million qubits for schoolbook.

The Karatsuba algorithm doesn't stack up well against the other algorithms for 2048-bit numbers, but, as is shown in Fig. 4, Karatsuba exhibits a saw-like behavior where there are large local variations in resources. Thus, Karatsuba could relatively perform better for other bit-sized inputs that are non-standard in RSA protocols.

V. Conclusion and Outlook

Azure Quantum Resource Estimator provides an alternative route to resource estimation of arithmetic primitives. Compared to research using the local resource estimation for the various multiplication algorithms, we were able to show that the new Azure Quantum Resource Estimator provides similar estimations [3]. Furthermore, when all three algorithms are compared on superconducing, trapped ion, and Majorana platforms, the windowed algorithm performs the best for all three platforms with Majorana being the most efficient.

These algorithms were initially designed to make classical multiplication more efficient. Therefore, the algorithms have the potential for further improvement for different qubit platforms by considering physical implementation, which would be the next step in comparing the arithmetic primitive algorithms.

References

[1] Haener, T., Soeken, M., Roetteler, M., Svore, K. M. "Quantum circuits for floating-point arithmetic" Reversible Computation: 10th International Conference, RC 2018, Leicester, UK, September 12-14, 2018, Proceedings 10. Springer International Publishing, 2018.

[2] Gidney, Craig. "Asymptotically efficient quantum Karatsuba multiplication." arXiv preprint arXiv:1904.07356 (2019).

[3] Gidney, Craig. "Windowed quantum arithmetic." arXiv preprint arXiv:1905.07682 (2019)

[4] Lopez, Sonia. "Customize resource estimates to machine characteristics." Microsoft, 15 March 2023, https://learn.microsoft.com/en-us/azure/quantum/overview-resources-estimator#output-data

[5] Roetteler, Martin, Michael Naehrig, Krysta M. Svore, and Kristin Lauter. "Quantum resource estimates for computing elliptic curve discrete logarithms." In Advances in Cryptology–ASIACRYPT 2017: 23rd International Conference on the Theory and Applications of Cryptology and Information Security, Hong Kong, China, December 3-7, 2017, Proceedings, Part II 23, pp. 241-270. Springer International Publishing, 2017.

[6] Beverland, M. E., Murali, P., Troyer, M., Svore, K. M., Hoeffler, T., Kliuchnikov, V., ... Vaschillo, A. (2022). Assessing requirements to scale to practical quantum advantage. arXiv preprint arXiv:2211.07629.