QUANTUM RESOURCE ESTIMATION OF ARITHMETIC PRIMITIVES

STUDENTS: Ethan Hansen, Sanskriti Joshi, Hannah Rarick

Motivation

As quantum computers progress towards a larger scale, it is imperative the "Top" of the computing-technology stack is improved. This project investigates the quantum resources required to compute primitive arithmetic algorithms. By using various quantum resource estimators, like Azure Quantum Resource Estimator , we can determine the resources required for numerous quantum algorithms [1]. In this project, we will provide a comprehensive resource analysis of numerous quantum multiplication algorithms such as Karatsuba, legacy, and windowed arithmetic for different qubit platforms (trapped ion, superconducting, Majorana).

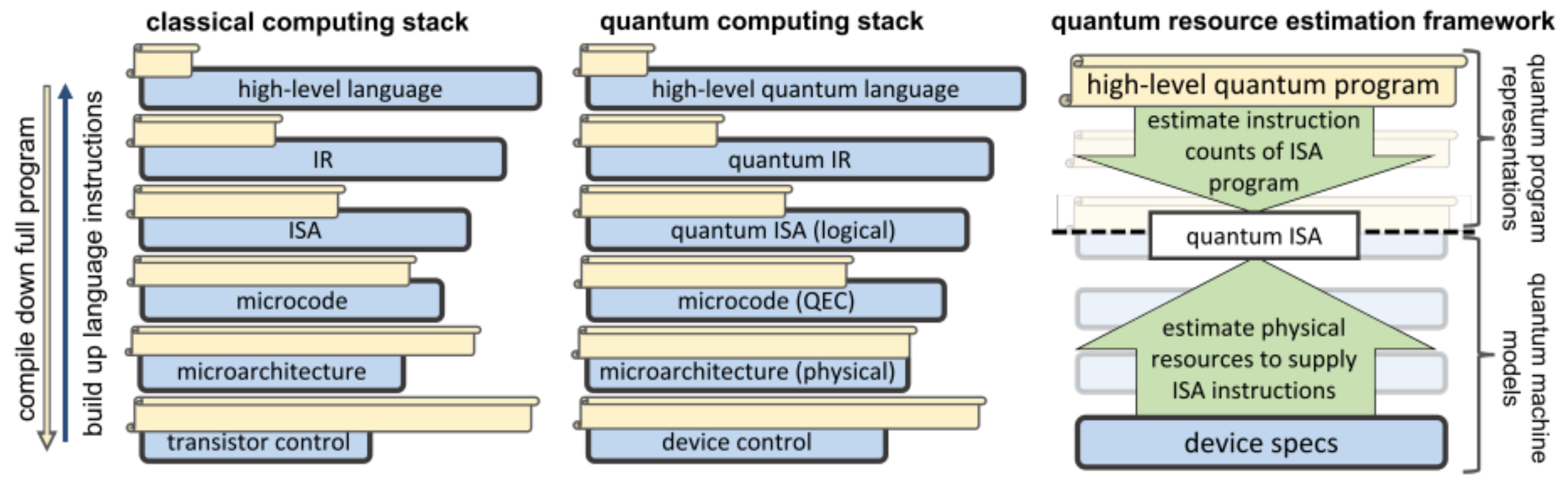
classical computing stack
high-level language
IR
ISA
microcode
microarchitecture
transistor control
compile down full program
build up language instructions
quantum computing stack
high-level quantum language
quantum IR
quantum ISA (logical)
microcode (QEC)
microarchitecture (physical)
device control
quantum resource estimation framework
high-level quantum program
estimate instruction counts of ISA program
quantum ISA
estimate physical resources to supply ISA instructions
device specs
quantum program representations
quantum machine models

Figure 1: The stacks for classical (left), quantum (center) computing, and quantum resource estimation framework (right) [1].

To ensure scalability of quantum systems it is crucial to understand the quantum resources required for various architecture design choices for a quantum stack (Fig. 1). Resource estimation provides different metrics to understand the approximate resources required to run a quantum algorithm on a specific hardware setup, including the number of qubits, quantum gates, time, etc. [2]. This can allow for an understanding of how different implementations of an algorithm can impact resource usage.

Methods

We refactored Q# code for implementing quantum multiplication using three different algorithms from the literature [3,4]:

- **Legacy Multiplication**
- **Karatsuba Multiplication**
  Improvement upon legacy multiplication discovered by Anatoly Karatsuba in 1960
- **Windowed Multiplication**
  A look-up table of precomputed values up to some set bit-size is classically constructed. The computation will be broken up into a sum of smaller bit-sized multiplications (just like Karatsuba and legacy) until all multiplication can be looked up. In the quantum form of this algorithm, the look-up table will be classically computed and then stored on the qubits. This creates more classical operations to perform, but decreases the overall quantum operations, which are much more costly and slower.

- Azure Quantum Resource Estimator accounts for both the qubit platform (gate-set, error-rates, gate-times) and the quantum error correction scheme (surface code or floquet code(Majorana systems only)),
- Allows one to submit a Q# circuit and gain information such as the required physical qubits and the algorithm runtime.

Results & Analysis

Platform Resource Comparison for 2048-bit Integers

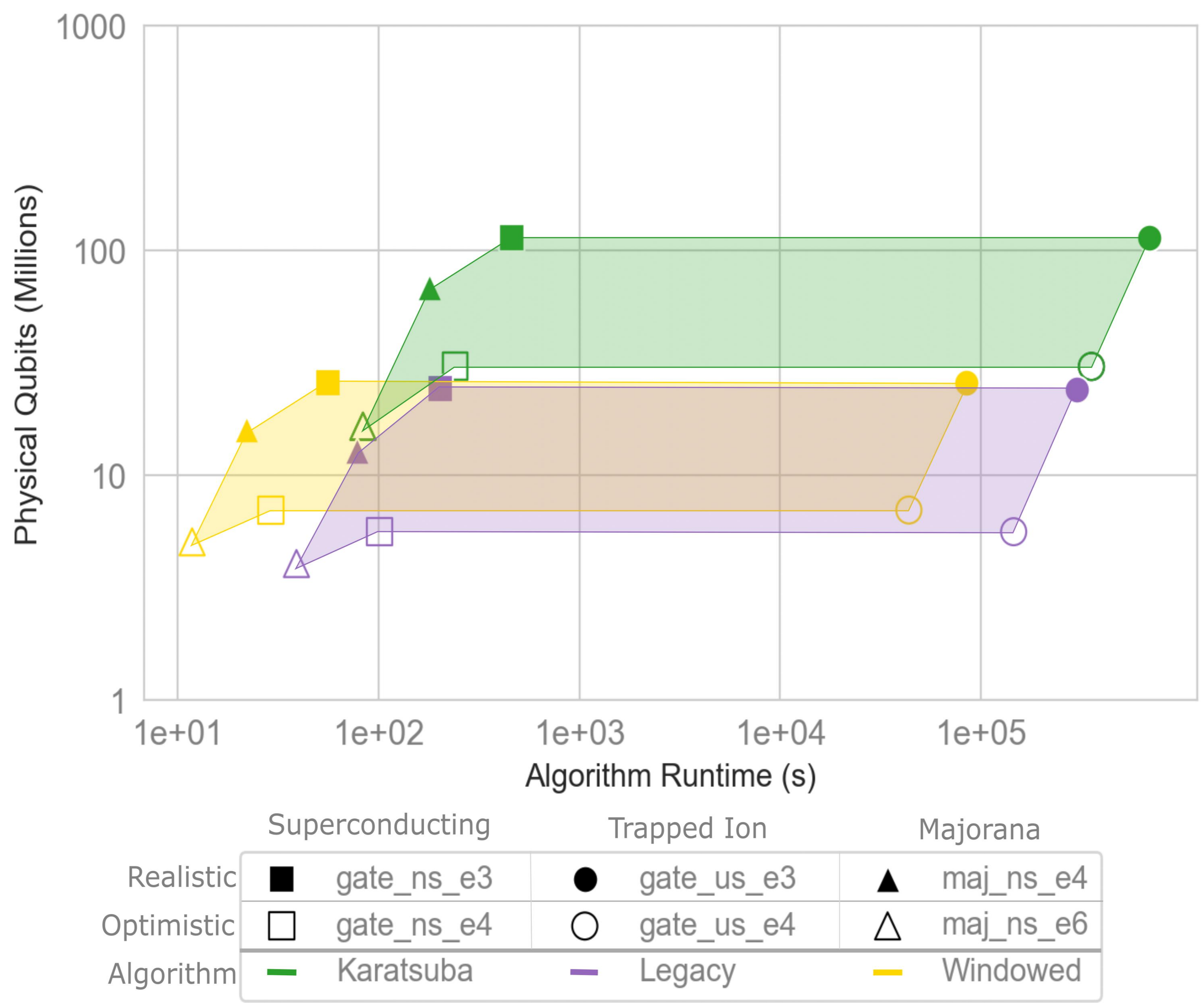

Figure 3: Estimated runtime and physical qubits required for multiplying two **2048-bit sized integers** using the **windowed (yellow)**, **legacy (purple)**, and **Karatsuba (green)** algorithms assuming a **superconducting (squares)**, **trapped ion (circles)**, or **Majorana (triangles)** qubit platform. The filled shapes represent a realistic qubit system that extrapolated from current error-rate estimates, while the open shapes represent an optimistic, higher fidelity qubit platforms.

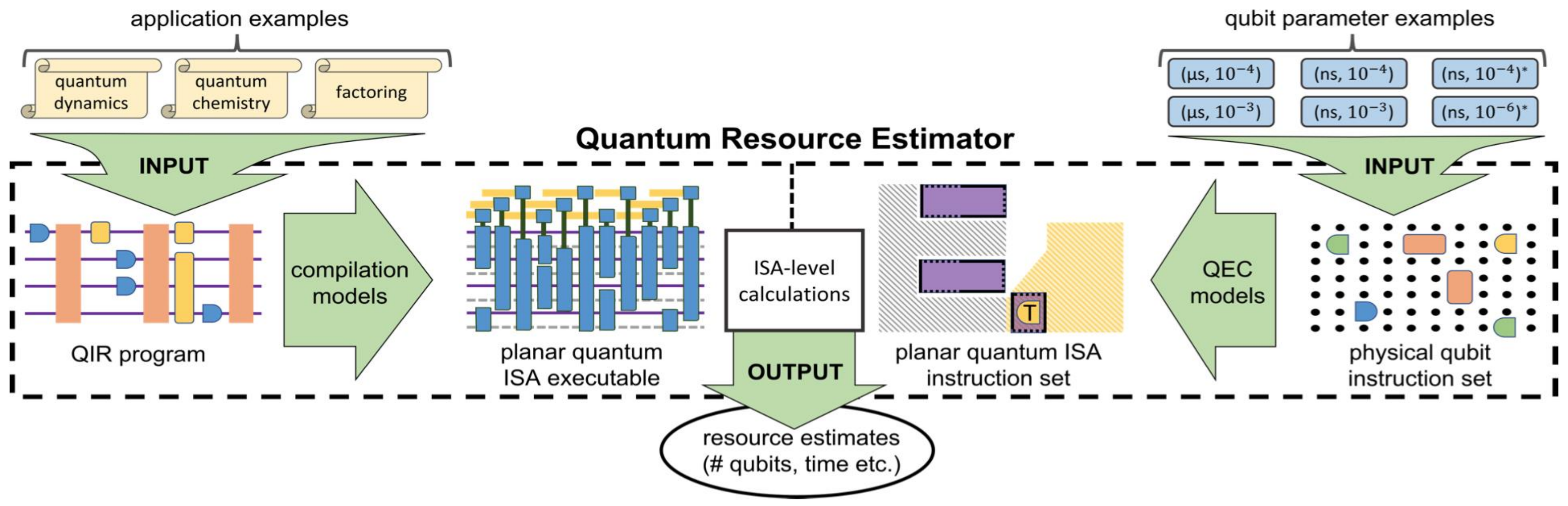

Figure 2: Block diagram of the Azure Quantum Resource Estimator [1]

Algorithm Resource Comparison

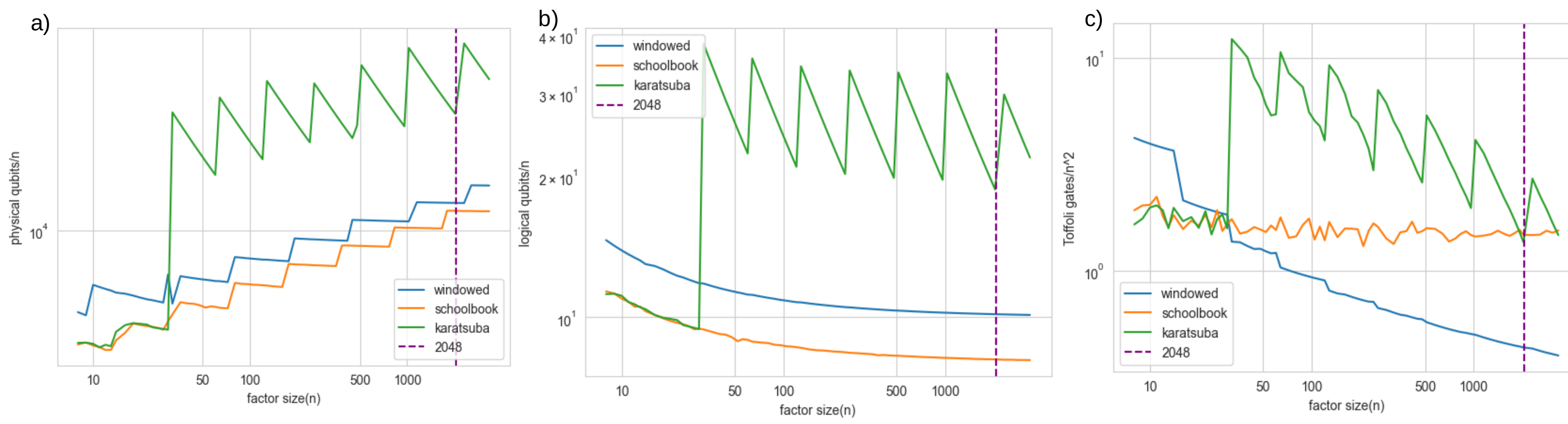

Figure 4: Estimated resources required for multiplying two n-bit sized integers using the windowed, schoolbook, and Karatsuba algorithms. **(a)** Estimated physical qubits divided by the factor size (n). **(b)** Estimated logical qubits divided by the factor size (n). **(c)** Estimated Toffoli gates divided by the squared factor size (n^2).

Algorithm Comparison Analysis

- Fig. 3 illustrates the runtime advantage for the windowed multiplication algorithm.
- In Fig. 4, the resources required relating to physical qubits, logical qubits, and Toffoli gates are compared for all three algorithms. The Toffoli gate is a controlled- controlled-NOT gate that can be decomposed into the gates of the Clifford+T gate set and is universal for reversible computing [5,6]. For higher input bit values, the Karatsuba algorithm becomes more efficient for Tofolli gates (Fig. 4c).
-

Conclusion & Future Work

Although Karatsuba, is theoretical an efficient algorithm, for larger and more interesting input sizes the windowed algorithm offers more advantageous runtimes (Fig. 3).
The final deliverable will be a comprehensive analysis of the resources required for the various quantum algorithms. This information will be within a Github repository containing the information required to reproduce the results for various quantum algorithms.

GitHub QR Code

References

[1] Beverland, Michael E., et al. "Assessing requirements to scale to practical quantum advantage." *arXiv preprint arXiv:2211.07629* (2022).
[2] 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
[3] Gidney, Craig. "Asymptotically efficient quantum Karatsuba multiplication." *arXiv preprint arXiv:1904.07356* (2019).
[4] Gidney, Craig. "Windowed quantum arithmetic." *arXiv preprint arXiv:1905.07682* (2019). → Windowed Arithmetic
[5] Roetteler, Martin, et al. "Quantum resource estimates for computing elliptic curve discrete logarithms." 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. Springer International Publishing, 2017.
[6] Selinger, Peter. "Quantum circuits of T-depth one." *Physical Review A* 87.4 (2013): 042302.

ELECTRICAL & COMPUTER ENGINEERING
UNIVERSITY *of* WASHINGTON

ADVISERS: Prof. Boris Blinov, Dr. Wim van Dam, Mariia Mykhailova

SPONSOR: MICROSOFT