CSE439 Fall 2025 Week 14: Quantum Advantage and Classical Catchup

Quantum Advantage refers to a benchmark achieved by quantum computing devices that classical devices cannot match. We have seen several examples in this course, of varying degrees of fairness-of-comparison, generality, and provenness:

- **Deutsch's Problem** and its **Deutsch-Jozsa extension** show that a quantum circuit can distinguish balanced Boolean functions from the constant-true and constant-false functions f using only 1 query $F_f(X)$ on a superposed state X, whereas classical circuits require more than one query $f(x)$ to do so. [But $F_f(X)$ to $f(x)$ may be an unfair parallel.]
- **Simon's Algorithm** computes the "hidden string" s of an n-ary Boolean function f that satisfies $f(x) = f(y) \iff x \oplus y = s$ for all $x, y \in \{0,1\}^n$, where it is proven that no classical randomized algorithm with query access to f can do so in time $n^{O(1)}$. [But this also has the "unfair parallel" and this is again a "promise problem", not a general computational problem.]
- **Shor's Algorithm** can factor n-digit numbers in "BQP-time" $\widetilde{O}(n^2)$ with high probability, where there are technical reasons to believe that every classical (randomized or deterministic) algorithm requires time $\Omega(2^{n^\epsilon})$ to do so, for some fixed $\epsilon > 0$ (where $\epsilon \sim 1/3$ is achievable and believed to be the limit, or maybe $\epsilon \sim 1/4$ or maybe $\epsilon \sim 1/5$). [But this advantage remains unproven, plus the algorithm makes high demands on precision.]
- The **CHSH Game** allows two quantum parties holding entangled states to achieve success rates over 85% in a game where unentangled parties can achieve 75% maximum. Success over 84% has been demonstrated in practice, and the 2022 Physics Nobel went to this. [But it is not a standalone general computational problem.]
- Team(s) led by Google have become more successful recently at defending and extending their 2019 claim (see section 14.8 and/or my article) of "quantum supremacy" for the problem of estimating the success probability of certain geometrically defined families of quantum circuits. [But this problem is "omphaloskeptic" and claims of advantage for general problems of interest such as solving large systems of equations are harder to pin down.]

Lipton and I ended our text with the question of where the quantum advantage comes from, looking at quantum circuits in particular. We fingered the asserted ability to create exponentially superposed states X like $\sum_x |x\rangle|f(x)\rangle$ (which is importantly different from $|+^n\rangle$) with high precision. But now I believe it is equally the property that quantum measurement can automatically "shout out" a string x that achieves a high local maximum of amplitude, whereas multivariable optimization is a classical hard problem area. Google's claim rests on exactly this.

As such, it is a fact of Nature---we can even see it with our own eyes. Nevertheless, we can talk about how well classical computing can play the ensuing game of "hide-and-seek". This is where the concept of **tensor networks** and applications of the SVD come in.

Can We Scale This Up?

Approximating entangled states by separable states---and telling properties of mixed states whether they are given as traceouts or not---goes into research that is plagued by **NP-hardness**. Bear in mind that the SVD representations have the same exponential "N" scaling as the underlying state vectors and matrices---as opposed to the order-n scaling of quantum circuits. Scott Aaronson makes these points pithily in his own notes. [Added: I mentioned the analogy between *tensor contraction* and *database join* further down. The paper https://arxiv.org/html/2209.12332v5, from October 2024, leverages this to show that even though certain problems of optimal contraction order are NP-hard, in the nice case of tree tensor networks and with a linearity condition, polynomial time algorithms are available. It also prominently references the dissertation work of Mahmoud Abo Khamis under Drs. Atri Rudra and Hung Ngo here at UB.]

Thus we cannot expect to be able to generate good approximations of arbitrary quantum states "given cold." This leaves two main possibilities as I see them:

1. Carry along succinct approximations to quantum states inductively as they are processed and built up in quantum circuits.
2. Focus on families of quantum states that have special structure that promotes classical approximations.

The main argument for 1 is evidently Nature computes efficiently, so has some way to avoid the exponential blowup that is ingrained in our explicit notation. Whether that applies to something as advanced as Shor's algorithm incurs other considerations---as the real-world quantum feasibility of Shor's algorithm is still not really established.

The rationale for 2 requires that the special structure does not impede the usefulness of quantum circuits/algorithms that abide by it. The major structural divide we have seen is between the Clifford family of gates: $\mathbf{H}$, $\mathbf{X}$, $\mathbf{Y}$, $\mathbf{Z}$, $\mathbf{S}$, $\mathbf{CNOT}$, $\mathbf{CZ}$, versus the fact that adding any one of the gates $\mathbf{T}$, $\mathbf{CS}$, $\mathbf{CCZ}$, or the Toffoli gate $\mathbf{CCX}$ gives the full power of quantum computation. The more fruitful structural limitations may apply to how gates are combined in circuits rather than which gates are allowed.

On the latter there is one major strand I know: Circuits that can be modeled as **tensor networks** that are close to being *trees* can be simulated classically with reasonable overhead. So we will say some words about tensor networks, as they are vital in classical machine learning as well.

A **tensor** T is a possibly higher-dimensional matrix. In the text's functional notation with tiered indexing, it is represented by a multi-ary function $T(i, j, k, \dots)$. The **order** is the number of tiers. In a **tensor network**, each tensor is a node of degree equal to its order. Edges, commonly called "legs", do not have to go to another node; they can be "free". Those that do go to another node (and so become a shared leg of the other node) represent setting up a **contraction**. The allowed operations in a tensor network include:

1. Introduce a new tensor---this is implicitly a tensor product with the existing tensors.
2. **Reshape** a tensor in a way that changes its order.
3. **Contract** two nodes along one or more shared legs. *Matrix product* is the canonical simple example. The generalized concept was employed by Einstein via the Einstein summation convention.

Matrix product states are a relatively simple case that suffices for one-dimensional arrays of n qubits---which are all we need for quantum circuits anyway. They do single out circuits whose gates act on neighboring qubits, but we can emulate those via the $\{\mathbf{H}, \mathbf{CS}\}$ basis along with swap gates, with 3-qubit gates like Toffoli on neighboring lines as an extra. Key ideas:

- Any n-qubit state can be represented as an order-n tensor $T[x_1, x_2, \cdots, x_n]$. Note that the commas are dimensions like in a matrix, not entries like in a vector---often we have skipped writing the commas to emphasize this.
- T can be written as an "expanding contraction" of order-3 tensors:
$$U_{a_1 x_1 b_1} U_{a_2 x_2 b_2} \cdots U_{a_n x_n b_n}.$$
- Here the x_i (which many sources write as σ_i which may be confusing with SVD notation but it will enter the picture) are the "physical dimensions" and range up to $N = 2^n$ for a binary (i.e., qubit) system. The other "virtual bonding dimensions" can reproduce T exactly if they are allowed to range up to N as well, but the point is to hold them down to a much smaller range k.

Using k-truncated SVDs for the U_i tensors gets the whole computation down to $N \times n$ rather than order N^2 as with density matrices and n-qubit operators. But there's still an N. Cutting that down depends on the usefulness of *succinct representations* for the objects---to which we apply SVD to begin with.

Segue to sources:

https://arxiv.org/abs/1306.2164 (A representative research-level survey from 2014.)

https://www.benasque.org/2020scs/talks_contr/106_tensornetworks_lecture1.pdf
(High-level but slides 15--20 are the best exposition of the SVD-based simulation idea in general contexts that I have found. Slides 19 and 20 address when and whether the SVD truncation represents the computed quantum states accurately enough.)

and especially

https://www.quantumcomputinglab.cineca.it/wp-content/uploads/2021/10/MPS_Lecture.pdf
(This is the most immediately accessible gateway to the main idea that I've found---i will go through much of it.)

https://pennylane.ai/qml/demos/tutorial_tn_circuits
(This shows how we could actually program this stuff---but I have not yet had time to do so by augmenting my existing C++ simulator, which I demo'ed for Shor's Algorithm.)

The current status as of end-2025 is that classical approaches, most prominently ones using MPS or tree-restricted forms of tensor networks, have not yet been left in the dust by physical quantum computers at relevant scale. I wrote a short poem to express the current state:

"It From Bit" we once proclaimed,
but now the Bit has bit the dust
of whizzing quantum chips that gamed
coherence to evade the trust
that the Word framed creation's hour
(Mother Nature fully lexical).
Why not evolve us that same power?
It is a status most perplexical.

The seventh line expresses the "kvetch" that if quantum advantage exists in Nature, why did 4+ billion years of evolution leave us with classical brains? Well, maybe not: Sir Roger Penrose's idea of quantum operations within the brain (see this) has recently come out a bit from behind a cloud it has been under since 1991.

The status that matters more for young careers is the viability of quantum computing science. Last month's visiting speaker Christopher Monroe wound up not being too-rosy from my point of view (whereas even my non-quantum colleagues recognize the likes of this as way overhyped): cautious on universal quantum computation and large-scale networking, but gung-ho on quantum devices and information-based applications. This is also partly why I've taken as much of a physical rather than purely-numerical view as I have in this course.