CSE491/596 Lecture Mon. Nov. 13: Quantum Computing I

Philosophy I: "Simple Realism"

- Show polarizing filters. (Link to chapter with photo.)
- Show part of talk https://cse.buffalo.edu/~regan/Talks/UnionCollege52115.pdf

Philosophy II: Is Nature *Lexical*?

- The idea of *Logos* from 500 BCE. Identified, perhaps incorrectly, with "word".
- The possible meaning of the final sentence of Umberto Eco's novel *The Name of the Rose,* quoting Bernard of Cluny, 1100s*:*

Stat rosa pristina nomine; nomina nuda tenemus

This means: *The [original] rose abides (as a/by its) [original/former] name; we hold the bare name.* It is possibly a misquote of "Stat *Roma*..." meaning that we (in the 1100s or 2000s) only know the glory of ancient Rome through recorded memory of it. I, however, subscribe to a deeper reading that treats "pristina" as meaning "unsullied" rather than "original" and takes some liberties with grammar:

The rose abides unsullied by a name; we hold only the bare name.

Regarding the rose as representing Nature, the issue is whether Nature's workings must be read as paying heed to the symbolic way we describe them. The (theoretically-)efficient quantum factoring algorithm is a real challenge to the idea that nature is symbolically mathematical.

Quantum States

[Note: I have edited the following to number from zero in "underlying co-ordinates" as in the text. This is different from how most linear algebra texts do it. It will however be conventional to number "quantum coordinates" from 1.] Natural systems can be modeled (inefficiently!?) by vectors

$$\mathbf{a} = \begin{bmatrix} a_0 \\ a_2 \\ a_3 \\ \vdots \\ a_i \\ \vdots \\ a_{N-1} \end{bmatrix}.$$

We say that $\mathbf{a}$ has N "underlying coordinates." Often N will be a power of 2, $N = 2^n$, where n will be the number of "quantum coordinates" or **qubits**. We can also have powers of larger numbers d, $N = d^n$. When $d = 3$ we will get **qutrits**, $d = 4$ will give **quarts**, and the general case gives **qudits**.

Maybe over 99% of the "QC" literature is about qubits. But actually, let's first think of N as not being subdivided at all.

One insight of linear algebra is that the entries a_i are not just "things unto themselves" but stand for multiples of corresponding **basis vectors**:

$$\mathbf{a} \;=\; a_0\,\mathbf{e_0} + a_1\,\mathbf{e_1} + a_2\,\mathbf{e_2} + \;\cdots\; + a_i\,\mathbf{e_i} + \;\cdots\; a_N\,\mathbf{e_N},$$

where for each i,

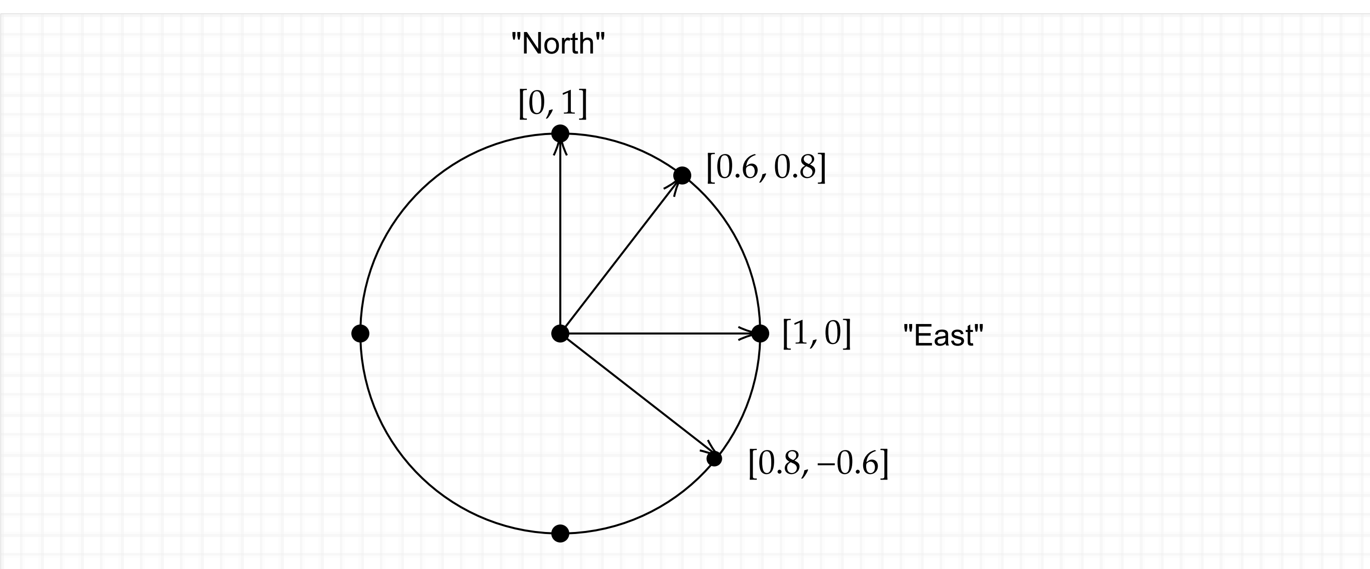
$$\mathbf{e_i} \;=\; [0, 0, 0, \,\ldots\,, 0, 1, 0, \,\ldots\,, 0]^T$$

with the lone 1 in position i. Notice we're being picky about considering vectors to be column vectors and writing transpose ^{T} to make $\mathbf{e_i}$ be a column vector. (Whether Nature really makes this distinction is a real question. We took the "no" side in the first edition, but using the angle-bracket notation from physics makes an initial commitment to the "yes" side.) With this notation, the vectors $\mathbf{e_i}$ are collectively called the **standard basis**.

A second insight of linear algebra is that one need not be "wedded to the standard basis"---one can do a **change-of-basis**. In general N-dimensional linear algebra, any set of N *linearly independent* vectors can be a basis. For instance, in $N = 2$ dimensions, the vectors

$[1, 0]$ and $[0.6, 0.8]$

are linearly independent (since there are only two vectors, the point is that neither is a multiple of the other). However, the second one is kind-of redundant in the first coordinate with the first. Whereas $\mathbf{e_0} \;=\; [1, 0]$ is "only East" and $\mathbf{e_1} \;=\; [0, 1]$ is "only North"---they are **orthogonal**, meaning that their *inner product* is zero.



We can diagram these vectors on the *unit circle*---note that $0.6^2 + 0.8^2 = 0.36 + 0.64 = 1$. The inner product of $[0.6, 0.8]$ and our "East" vector is $0.6 \cdot 1 + 0.8 \cdot 0 = 0.6$.

There are several ways to write the inner product of two vectors $\mathbf{a}$ and $\mathbf{b}$:

$$\mathbf{a} \cdot \mathbf{b}, \quad \langle \mathbf{a}, \mathbf{b} \rangle, \quad \langle \mathbf{a} \mid \mathbf{b} \rangle.$$

The last is what feeds into **Dirac Notation**, as the **bra**(c)**ket** of the row vector $\langle \mathbf{a}|$ and the column vector $|\mathbf{b}\rangle$. I will introduce this notation later---referencing chapter 14 of my text. But first, we'll cover an aspect of linear algebra that isn't really covered in Math 309 here: **tensor products** and *what they mean*.

Tensor Products

How Does Nature Compute?

$\Downarrow$

How Does Nature Concatenate?

Strings: Trivial operation: $a \cdot b = ab$

$x \cdot y = xy$

Nature uses (linear) operators $\quad A \cdot B = \{x \cdot y : x \in A \wedge y \in B\}$

that we represent as matrices (over a standard basis).

When you think of matrices and vectors, the first idea that pops into mind is the ordinary matrix product AB of an $\ell \times m$ and an $m \times n$ matrix. But this is "lossy," whereas concatenation must be lossless (except possibly for memory of the place where the strings got concatenated). Instead, Nature uses **tensor product**, which applies also to vectors and doesn't need the "shapes" of the operands to agree.

[I then did pen and paper material that mostly included the following examples, except that I drew them as column vectors.]

CSE 491/596 Lecture 11/13/23

Tensor Product of Two 2-dim. vectors.

$$\underbrace{\begin{bmatrix}a_1\\a_2\end{bmatrix}}_{a} \otimes \underbrace{\begin{bmatrix}b_1\\b_2\end{bmatrix}}_{b} \qquad \text{“}a\otimes b\text{”} \begin{bmatrix} a_1 \cdot \begin{bmatrix}b_1\\b_2\end{bmatrix} \\ a_2 \cdot \begin{bmatrix}b_1\\b_2\end{bmatrix}\end{bmatrix} = \begin{bmatrix}a_1b_1\\a_1b_2\\a_2b_1\\a_2b_2\end{bmatrix}$$

$|0\rangle \otimes |0\rangle$

$$\begin{bmatrix}1\\0\end{bmatrix} \cdot \begin{bmatrix}1\\0\end{bmatrix} = \begin{bmatrix} 1 \cdot \begin{bmatrix}1\\0\end{bmatrix} \\ 0 \cdot \begin{bmatrix}1\\0\end{bmatrix}\end{bmatrix} = \begin{bmatrix}1\\0\\0\\0\end{bmatrix} \text{ “}e_0\text{” } |00\rangle$$

$|0\rangle \otimes |1\rangle$

$$\begin{bmatrix}1\\0\end{bmatrix} \otimes \begin{bmatrix}0\\1\end{bmatrix} = \begin{bmatrix} 1 \begin{bmatrix}0\\1\end{bmatrix} \\ 0 \begin{bmatrix}0\\1\end{bmatrix}\end{bmatrix} = \begin{bmatrix}0\\1\\0\\0\end{bmatrix} \quad \text{“}e_1\text{”} = |01\rangle$$

$$\begin{bmatrix}0\\1\end{bmatrix} \otimes \begin{bmatrix}1\\0\end{bmatrix} = \begin{bmatrix} 0 \begin{bmatrix}1\\0\end{bmatrix} \\ 1 \begin{bmatrix}1\\0\end{bmatrix}\end{bmatrix} = \begin{bmatrix}0\\0\\1\\0\end{bmatrix}$$

$|1\rangle \otimes |0\rangle = |10\rangle = e_2$

$$\begin{bmatrix}0\\1\end{bmatrix} \otimes \begin{bmatrix}0\\1\end{bmatrix} = \begin{bmatrix}0\\0\\0\\1\end{bmatrix}$$

$|1\rangle \otimes |1\rangle$

$|11\rangle =$ “e_3”

Big-Endian ← ✓ Our text
or
Little Endian?

should $\begin{bmatrix}1\\0\end{bmatrix}$ stand for “0” ✓
$\begin{bmatrix}0\\1\end{bmatrix}$ for “1” or vice-versa?

zero 0 in binary is 00
1 in binary is 01
2 in binary is 10
3 in binary is 11

Big-Endian: $[1\,0\,0\,0]^T$ comes first
Little Endian: opposite

[Here is a typeset version of most of this, too---making things look like row vectors instead.]

An n-qubit quantum state is denoted by a unit vector in $\mathbb{C}^N$ where $N = 2^n$. Thus, a 2-qubit state is represented by a unit vector in $\mathbb{C}^4$. That takes up 8 real dimensions, and even trying tricks as for the Bloch Sphere would bring that down only to a 6-dimensional hypersurface in $\mathbb{R}^7$. Until we have a Hyper-Zoom able to help us visualize 7-dimensional space, we have to rely on linear algebra and some general ideas shared by Hilbert Spaces whether real or complex.

One of those ideas is the **standard basis**. In 4-space, this is given by the vectors:

$$e_0 = (1,0,0,0),\ e_1 = (0,1,0,0),\ e_2 = (0,0,1,0),\ e_3 = (0,0,0,1).$$

The indexing scheme for **quantum coordinates** changes the labels to come from $\{0,1\}^2$ instead of from $\{1,2,3,4\}$, using the canonical binary order $00, 01, 10, 11$. Then we have:

$$e_{00} = (1,0,0,0),\ e_{01} = (0,1,0,0),\ e_{10} = (0,0,1,0),\ e_{11} = (0,0,0,1).$$

The big advantage is that these basis elements are all separable and the labels respect the tensor products involved:

$$\begin{array}{lllllll}
|00\rangle = e_{00} &= (1,0,0,0) &= (1,0)\otimes(1,0) &= e_0 \otimes e_0 &= |0\rangle\otimes|0\rangle &= |0\rangle|0\rangle \\
|01\rangle = e_{01} &= (0,1,0,0) &= (1,0)\otimes(0,1) &= e_0 \otimes e_1 &= |0\rangle\otimes|1\rangle &= |0\rangle|1\rangle \\
|10\rangle = e_{10} &= (0,0,1,0) &= (0,1)\otimes(1,0) &= e_1 \otimes e_0 &= |1\rangle\otimes|0\rangle &= |1\rangle|0\rangle \\
|11\rangle = e_{11} &= (0,0,0,1) &= (0,1)\otimes(0,1) &= e_1 \otimes e_1 &= |1\rangle\otimes|1\rangle &= |1\rangle|1\rangle
\end{array}$$

It is OK to picture the tensoring with row vectors, but because humanity chose to write matrix-vector products as Mv rather than vM, they need to be treated as column vectors. This will lead to cognitive dissonance when we read quantum circuits left-to-right but have to compose matrices right-to-left. Lipton and I are curious whether a "non-handed" description of nature can work.