



記法と演算

記譜法

音の特徴を表現するよ

形は音の長さ

音符	○	♪	♩	♪	♩
音の長さ	4	2	1	1/2	1/4

縦方向は音の高さ



横方向は音の流れ

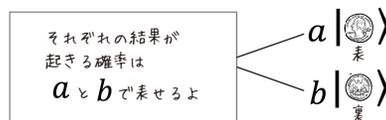


ディラックの記法

ブラケット記法

複雑な量子状態を表すよ

それぞれの起こりうる結果は「ケット」の中に書く → $| \rangle$



$$|a|^2 \leftarrow \text{結果が表か裏になる確率はそれぞれ} \rightarrow |b|^2$$

公平なコインだと、表と裏の確率は同じだね

$$\frac{1}{\sqrt{2}} | \text{表} \rangle + \frac{1}{\sqrt{2}} | \text{裏} \rangle$$

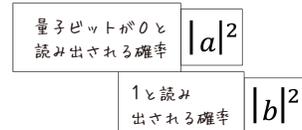
量子の記法

複雑な量子状態を表すよ

起こりうる結果は0と1だね

$$a|0\rangle + b|1\rangle \leftarrow \text{1量子ビット}$$

確率を表す!



$$\text{注: } |a|^2 + |b|^2 = 1$$

この量子状態のとき、0と1がそれぞれ測定される確率はいくつかな?

$$\frac{1}{\sqrt{2}} |0\rangle + \frac{1}{\sqrt{2}} |1\rangle$$

0と1がそれぞれ1/2の確率で読み出されるよ

$$\frac{2}{1} = 2q \quad \frac{2}{1} = 2p \quad : \text{? 量}$$

量子の略記法

量子の記法には、代数みたいに見やすくするための慣習があるよ

$$1x + 0y = 15$$

$$\downarrow$$

$$x = 15$$

量子の記法でも同じように略しちゃおう

$$1|0\rangle + 0|1\rangle \Rightarrow |0\rangle$$

同じ大きさの係数を省いちゃうこともあるよ

$$\frac{1}{\sqrt{2}}|0\rangle + \frac{1}{\sqrt{2}}|1\rangle \Rightarrow |0\rangle + |1\rangle$$

行列での記法

これでも量子状態が表せる!

量子ビットの状態は、行列でも記述できるよ



ディラックの記法 $\frac{1}{\sqrt{2}}|0\rangle + \frac{1}{\sqrt{2}}|1\rangle$

行列での記法 $\frac{1}{\sqrt{2}} \begin{bmatrix} 1 \\ 1 \end{bmatrix}$

どっちの記法も、量子ビットを0と読むか1と読むか、その確率は半々、ということを表しているよ

線形代数

行列の乗算が、ゲート演算するのに使われるよ

例えば、

\boxed{X} ゲートの演算は

$$\text{右の行列で表すんだ} \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$$

Xゲートを $|1\rangle$ (もしくは $|0\rangle$) に作用させるときは

結果は行列の乗算で求められるよ

$$\begin{matrix} |1\rangle & \boxed{X} & |0\rangle \\ \text{前} & & \text{後} \end{matrix}$$

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

Xゲート ↓ |0> 前 |1> 後

もし \boxed{X} を次の量子ビットに作用させるなら: $\frac{\sqrt{2}}{2}|0\rangle + \frac{1}{2}|1\rangle$

量子ビットの状態はどうなる?

$$\begin{bmatrix} \frac{\sqrt{2}}{2} \\ \frac{1}{2} \end{bmatrix} = \begin{bmatrix} \frac{\sqrt{2}}{2} \\ \frac{1}{2} \end{bmatrix} \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$$

量子コンピューティングについて もっと知りたいならこちら

<https://www.epiqc.cs.uchicago.edu/resources/>

May 2023

Translated by QSCS, Kyushu University, Japan

This work is funded in part by EPiQC, an NSF Expedition in Computing, under grant 1730449

