量子計算ミニマム

筒井翔一朗

2022 年 2 月 4 日

概要

これは、ものすごい勢いで量子計算について学ぶためのノートです。

目次

1 overview

N スピン系を考える。これを古典的に扱う場合、状態ベクトルは N 次元。量子的には 2^N 次元。古典計算機で扱おうとすると、メモリも計算量も大変なことになってしまう。例えば、$N=28$ の場合、$2^{28}*(8*2) \sim 4.3\times 10^9$ byte. 量子コンピュータでは 2^N の自由度を直に扱うので、量子系のシミュレーションに向いている。現在実現しているのは 100bit の量子コンピュータ。もう十分では？ と思いきや、エラー訂正を実装しようとすると、100 万以上必要量子ムーアの法則が成り立つとすると、14 年後には 100 万 bit...

NISQ(Noisy Intermediate-Scale Quantum) computer

FTQC(Fault Tolerant Quantum Computer)

2 Notation

Qubit

$$|0\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \quad |1\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \tag{1}$$

Kronecker 積

$$A \otimes B \equiv \begin{pmatrix} a_{11}B & \dots & a_{1N}B \\ \vdots & \ddots & \vdots \\ a_{N1}B & \dots & a_{NN}B \end{pmatrix} \tag{2}$$

Pauli ゲート

$$X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \quad Y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}, \quad Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} \tag{3}$$

Hadamard ゲート

$$H = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \tag{4}$$

一般位相ゲート

$$R_l = \begin{pmatrix} 1 & 0 \\ 0 & e^{i\frac{2\pi}{2^l}} \end{pmatrix} \tag{5}$$

整数 k の 2 進数表記

$$(k)_2 = i_1 i_2 \cdots, \quad i_n = 0, 1 \tag{6}$$

例

$$(0)_2 = 0 \tag{7}$$
$$(1)_2 = 1 \tag{8}$$
$$(2)_2 = 10 \tag{9}$$
$$(3)_2 = 11 \tag{10}$$
$$(4)_2 = 100 \tag{11}$$
$$(5)_2 = 101 \tag{12}$$
$$(6)_2 = 110 \tag{13}$$
$$(7)_2 = 111 \tag{14}$$
$$(8)_2 = 1000 \tag{15}$$
$$(9)_2 = 1001 \tag{16}$$
$$(10)_2 = 1010 \tag{17}$$
$$(11)_2 = 1011 \tag{18}$$
$$(12)_2 = 1100 \tag{19}$$
$$(13)_2 = 1101 \tag{20}$$

小数を含む 2 進数表記

$$(k)_2 = k_1 \cdots k_{l-1}.k_l \cdots k_n = \cdots + k_{l-1}2^0 + \frac{k_l}{2^1} + \frac{k_{l+1}}{2^2} + \cdots + \frac{k_n}{2^{n-l+1}} \tag{21}$$

$(k)_2 = k_1 k_2 \cdots k_n$ のとき

$$k = k_1 2^{n-1} + k_2 2^{n-2} + \cdots + k_n 2^0 \tag{22}$$

例えば、

$$9 = 1 \times 2^3 + 0 \times 2^2 + 0 \times 2^1 + 1 \times 2^0 \tag{23}$$

3　測定

3.1　射影測定

A を物理量、すなわちエルミート演算子とする。A の固有値と固有ベクトルをそれぞれ a, $|a\rangle$ とする。a は離散的な値を取るものとする。

$$A|a\rangle = a|a\rangle \tag{24}$$

量子力学における測定とは、A の固有値を測る行為のことである。一般に、測定のたびに得られる固有値は異なり、状態 $|\psi\rangle$ に対して A を測定して、固有値 a が得られる確率は

$$P_a = |\,|a\rangle\langle a|\psi\rangle\,|^2 \tag{25}$$

で与えられる。測定が行われた後の状態は

$$\frac{1}{|\,|a\rangle\langle a|\psi\rangle\,|}|a\rangle\langle a|\psi\rangle \tag{26}$$

となる。

例えば、状態

$$|\psi\rangle = \alpha\,|0\rangle + \beta\,|1\rangle\,, \quad |\alpha|^2 + |\beta|^2 = 1 \tag{27}$$

に対して、

$$Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} \tag{28}$$

の固有値を測定することを考える。$|0\rangle$, $|1\rangle$ は Z の固有状態である。

$$Z\,|0\rangle = |0\rangle\,, \quad Z\,|1\rangle = -\,|1\rangle \tag{29}$$

よって、測定値が 1 である確率は

$$P_01 = |\,|0\rangle\,\langle 0|\psi\rangle\,|^2 = |\,\langle 0|\psi\rangle\,|^2 = |\alpha|^2 \tag{30}$$

であり、測定値が -1 である確率は

$$P_{-1} = |\beta|^2 \tag{31}$$

である。この測定は頻繁に登場するが、$|\psi\rangle$ が $|0\rangle$ or $|1\rangle$ のどっちになっているかを測定する、という言い方をする場合がある。

また、状態空間がテンソル積で与えられていて、部分系のみを測定するということも可能である。状態 $|\psi\rangle \otimes |\psi'\rangle$ に対して、1 番目の部分系に作用する演算子 A を測定して、固有値 a が得られる確率は

$$P_a = |(|a\rangle\,\langle a| \otimes I)\,|\psi\rangle \otimes |\psi'\rangle\,|^2 = |\,|a\rangle\,\langle a|\psi\rangle \otimes |\psi'\rangle\,|^2 \tag{32}$$

で与えられる。

3.2 POVM

ここでは有限次元の Hilbert 空間のみ考える。半正定値行列のセット $\{F_i\}$ で

$$\sum_i F_i = I \tag{33}$$

を満たすようなものを positive operator-valued measure (POVM) と呼ぶ。量子状態 ρ を測定して、出力 i が得られる確率は

$$\mathrm{tr}(\rho F_i) \tag{34}$$

で与えられる。

例として、純粋状態

$$\rho = |\psi\rangle\,\langle\psi|\,, \quad |\psi\rangle = \frac{\alpha}{|\alpha|^2 + |\alpha|^2}\,|0\rangle + \frac{\beta}{|\alpha|^2 + |\alpha|^2}\,|1\rangle \tag{35}$$

と、POVM

$$F_0 = \frac{1+Z}{2} = \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix}, \quad F_1 = \frac{1-Z}{2} = \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix}, \tag{36}$$

を考える。測定によって出力 0 が得られる確率は

$$\mathrm{tr}(\rho F_0) = \mathrm{tr}(|\psi\rangle\langle\psi| F_0) = \langle\psi| F_0 |\psi\rangle = \frac{\alpha^2}{|\alpha|^2 + |\alpha|^2} \tag{37}$$

出力 1 が得られる確率は

$$\mathrm{tr}(\rho F_1) = \mathrm{tr}(|\psi\rangle\langle\psi| F_1) = \langle\psi| F_1 |\psi\rangle = \frac{\beta^2}{|\alpha|^2 + |\alpha|^2} \tag{38}$$

である。これはいわゆる Born の規則である。

4　量子ビットと演算

古典計算機では、$0, 1$ の 2 値を取りうる bit たちと、NOT, AND, OR などの論理演算を電子回路を用いて実装でき、これらを駆使して様々なアルゴリズムを構成している。一方量子計算機では、bit の役割を果たすのは内部自由度が 2 の量子論的な粒子である。これを qubit と呼ぶ。qubit の振る舞いは量子力学によって記述される。

1. ひとつの qubit の状態は、$\mathbb{C}$ 上の 2 次元 Hilbert 空間の元で表すことができる。
2. 複数の qubit がある場合はそれらのテンソル積で表現される。(例えば、N 個の qubit 系は 2^N 次元の Hilbert 空間の元である。)
3. qubit 系が孤立しているとき、qubit 系の時間変化はユニタリー変換を用いて記述される。

量子計算機では、qubit 系にユニタリー変換を繰り返し作用させることで、所望のアルゴリズムを構成する。ユニタリー変換しか許さないというのは強力な縛りで、古典計算機とはずいぶん様子が異なる。(例えば AND を考えてみよ。)

古典的な計算においては、任意の論理関数が NOT と AND の組み合わせで表現できることが証明でき、このとき組 {NOT, AND} は万能であるというのだった。このような性質のために、どんなに複雑な操作も、単純な論理演算の組み合わせで表現することができる。これは実機を作る上で重要である。量子計算にも同様の性質がある。

4.1　1 bit ユニタリー変換

ここでは 1 qubit 系とそれに作用するユニタリー変換を考える。1 qubit の状態は一般に、

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle \tag{39}$$

と表すことができる。ここで、

$$|0\rangle = \begin{pmatrix}1\\0\end{pmatrix}, \quad |1\rangle = \begin{pmatrix}0\\1\end{pmatrix}, \tag{40}$$

である。これらを計算機基底と呼ぶ。また、α, β は $|\alpha|^2 + |\beta|^2 = 1$ を満たす複素数で、しばしば

$$\alpha = \cos\frac{\theta}{2}, \quad \beta = e^{i\phi}\sin\frac{\theta}{2} \tag{41}$$

とパラメトライズする。θ, ϕ は実数である。これらを指定すると、半径 1 の球面上の一点が指定される。この球面のことを Bloch 球と呼ぶ。1 qubit 状態に作用するユニタリー変換は、サイズ 2 の行列で表現することができる。

$$U = \begin{pmatrix} u_{11} & u_{12} \\ u_{21} & u_{22} \end{pmatrix}, \quad u_{ij} \in \mathbb{C} \tag{42}$$

（この独立なパラメータの数は 4）ユニタリー変換は、Bloch 球の点を移す変換になっているはずである。x, y, z 軸周りの θ 回転は

$$e^{-i(\theta/2)A} = \cos\frac{\theta}{2}I - i\sin\frac{\theta}{2}A, \quad A = X, Y, Z \tag{43}$$

で表される。ここで、

$$X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \quad Y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}, \quad Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} \tag{44}$$

である。これらを Pauli 行列（ゲート）と呼び、以下の性質がある。

$$X^2 = Y^2 = Z^2 = I \tag{45}$$

$$XY - YX = iZ, \quad YZ - ZX = iX, \quad ZX - XZ = iy, \tag{46}$$

$$XY + YX = YZ + ZY = ZX + XZ = 0 \tag{47}$$

実際例えば、

$$e^{-i(\theta'/2)Z}|\psi\rangle = \left(\cos\frac{\theta'}{2}I - i\sin\frac{\theta'}{2}Z\right)\left(\cos\frac{\theta}{2}|0\rangle + e^{i\phi}\sin\frac{\theta}{2}|1\rangle\right) \tag{48}$$

$$= \cos\frac{\theta'}{2}\cos\frac{\theta}{2}|0\rangle + \cos\frac{\theta'}{2}e^{i\phi}\sin\frac{\theta}{2}|1\rangle - i\sin\frac{\theta'}{2}\cos\frac{\theta}{2}|0\rangle + i\sin\frac{\theta'}{2}e^{i\phi}\sin\frac{\theta}{2}|1\rangle \tag{49}$$

$$= e^{-i\theta'/2}\cos\frac{\theta}{2}|0\rangle + e^{i\theta'/2}e^{i\phi}\sin\frac{\theta}{2}|1\rangle \tag{50}$$

$$= e^{-i\theta'/2}\left(\cos\frac{\theta}{2}|0\rangle + e^{i\phi+\theta'}\sin\frac{\theta}{2}|1\rangle\right) \tag{51}$$

などにより、全体の位相を除いて確かに回転になっていることが確認できる。よって一般に、

$$U = e^{i\alpha}e^{-i(\beta/2)Z}e^{-i(\gamma/2)Y}e^{-i(\delta/2)Z} \tag{52}$$

のように表すことができる。β, γ, δ は Euler 角である。（パラメターが 4 つ）

このような U を少数のユニタリー変換の積で表すことができるのだろうか？ 実は次が示せる。

- H, T の積によって、任意の U をいくらでも精度良く近似できる。

ここで、

$$H = \frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}, \quad T = e^{-i(\pi/8)Z} \tag{53}$$

である。H を Hadamard ゲートと呼ぶ。これを示そう。まず Hadamard ゲートについて

$$H^{-1} = H, \quad X = HZH \tag{54}$$

が成り立つことに注意すれば、

$$HTH = He^{-i(\pi/8)Z}H = H\left(\cos\frac{\pi}{8}I - i\sin\frac{\pi}{8}Z\right)H = \cos\frac{\pi}{8}I - i\sin\frac{\pi}{8}X = e^{-i(\pi/8)X} \tag{55}$$

である。ここで、$V = THTH$ を考え、これがある軸回りの回転を引き起こすことを見る。一般に、$\vec{n} = (n_x.n_y, n_z)$ 軸回りの θ 回転は

$$e^{-i\theta\vec{n}\cdot\vec{\sigma}/2} = 1 + \left(-i\frac{\theta}{2}\vec{n}\cdot\vec{\sigma}\right) + \frac{1}{2}\left(-i\frac{\theta}{2}\vec{n}\cdot\vec{\sigma}\right)^2 + \frac{1}{3!}\left(-i\frac{\theta}{2}\vec{n}\cdot\vec{\sigma}\right)^3 + \cdots \tag{56}$$

$$= 1 - i\left(\frac{\theta}{2}\right)\vec{n}\cdot\vec{\sigma} - \frac{1}{2}\left(\frac{\theta}{2}\right)^2 + i\frac{1}{3!}\left(\frac{\theta}{2}\right)^3\vec{n}\cdot\vec{\sigma} + \cdots \tag{57}$$

$$= \cos\frac{\theta}{2}I - i\sin\frac{\theta}{2}(n_xX + n_yY + n_zZ) \tag{58}$$

と表されることに注意する。ただしここで、

$$(\vec{n}\cdot\vec{\sigma})^2 = (n_xX + n_yY + n_zZ)^2 \tag{59}$$

$$= n_x^2X^2 + n_y^2Y^2 + n_z^2Z^2 + n_xn_y(XY + YX) + n_yn_z(YZ + ZY) + n_zn_x(ZX + XZ) \tag{60}$$

$$= I \tag{61}$$

を用いた。

$$V = e^{-i(\pi/8)Z}e^{-i(\pi/8)X} \tag{62}$$

$$= \left(\cos\frac{\pi}{8}I - i\sin\frac{\pi}{8}Z\right)\left(\cos\frac{\pi}{8}I - i\sin\frac{\pi}{8}X\right) \tag{63}$$

$$= \cos^2\frac{\pi}{8}I - \sin^2\frac{\pi}{8}ZX - i\cos\frac{\pi}{8}\sin\frac{\pi}{8}Z - i\cos\frac{\pi}{8}\sin\frac{\pi}{8}X \tag{64}$$

$$= \cos^2\frac{\pi}{8}I - \sin^2\frac{\pi}{8}(iY) - i\cos\frac{\pi}{8}\sin\frac{\pi}{8}Z - i\cos\frac{\pi}{8}\sin\frac{\pi}{8}X \tag{65}$$

$$= \cos^2\frac{\pi}{8}I - i\sin\frac{\pi}{8}\left(\cos\frac{\pi}{8}X + \sin\frac{\pi}{8}Y + \cos\frac{\pi}{8}Z\right) \tag{66}$$

である。よって、V はある軸回りの回転で、その角度は

$$\theta = 2\arccos\frac{2+\sqrt{2}}{4} = \arccos\frac{2\sqrt{2}-1}{4} \tag{67}$$

である。これは π の無理数倍だから、V を何度もかけると、この軸周りの任意の回転を任意の精度で近似できる。次に $V' = HVH$ を考えると、また別の軸の周りの回転について同様のことが言える。以上を組み合わせることで証明終了。

ちなみに、θ が π の無理数倍であることは、$e^{2\pi i\theta}$ が $x^4 + x^3 + \frac{1}{4}x^2 + x + 1$ という円分多項式でない多項式のの根であることから従う。P. O. Boykin, T. Mor, M. Pulver, V. Roychowdhury, and F. Vatan: arXiv:quant-ph/9906054 また、Solovay-Kitaev により、近似に必要なゲートの数は高々多項式オーダーであることが示されている。

4.2　2 level ユニタリー変換

1 bit に作用するユニタリー変換はサイズ 2 の行列であった。実は、どんなユニタリー変換も、実質サイズ 2 の行列の積で表すことができる。例として、

$$U = \begin{pmatrix} a & * & * \\ b & * & * \\ * & * & * \end{pmatrix} \tag{68}$$

を考える。ここで、2-level のユニタリー行列として

$$U_1 = \frac{1}{\sqrt{|a|^2 + |b|^2}} \begin{pmatrix} a^* & b^* & 0 \\ b & -a & 0 \\ 0 & 0 & 1 \end{pmatrix} \tag{69}$$

というものを考える。これがユニタリーであることは

$$U_1^{-1} = \frac{1}{\sqrt{|a|^2 + |b|^2}} \begin{pmatrix} a & b & 0 \\ b^* & -a^* & 0 \\ 0 & 0 & 1 \end{pmatrix} \tag{70}$$

から分かる。U_1 を U にかけると

$$U_1 U = \begin{pmatrix} * & * & * \\ 0 & * & * \\ * & * & * \end{pmatrix} \tag{71}$$

のようにある成分が 0 になる。同じようなことをもう一度やると、

$$U_2 U_1 U = \begin{pmatrix} 1 & * & * \\ 0 & c & d \\ 0 & e & f \end{pmatrix} \tag{72}$$

の形にできる。$U_2 U_1 U$ はユニタリーだからこの段階で自動的に

$$U_2 U_1 U = \begin{pmatrix} 1 & 0 & 0 \\ 0 & c & d \\ 0 & e & f \end{pmatrix} \tag{73}$$

となっているはずである。最後に

$$U_3 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & c & d \\ 0 & e & f \end{pmatrix}^{-1} \tag{74}$$

とおけば、$U_3 U_2 U_1 U = I$ となる。よって、$U = U_1^\dagger U_2^\dagger U_3^\dagger$ と分解できた。もっとサイズの大きな U についても同様の分解が可能である。

4.3　2 bit ユニタリー変換

前節の議論から、任意の 2 つの成分に作用するユニタリー変換が構成できれば十分だということが分かった。2 bit 系の場合についてこれが可能なことを示そう。2 bit 系の基底は $|00\rangle, |01\rangle, |10\rangle, |11\rangle$ である。具体

的に計算してみると、

$$|00\rangle = \begin{pmatrix}1\\0\end{pmatrix} \otimes \begin{pmatrix}1\\0\end{pmatrix} = \begin{pmatrix}1\\0\\0\\0\end{pmatrix} \tag{75}$$

$$|01\rangle = \begin{pmatrix}1\\0\end{pmatrix} \otimes \begin{pmatrix}0\\1\end{pmatrix} = \begin{pmatrix}0\\1\\0\\0\end{pmatrix} \tag{76}$$

$$|10\rangle = \begin{pmatrix}0\\1\end{pmatrix} \otimes \begin{pmatrix}1\\0\end{pmatrix} = \begin{pmatrix}0\\0\\1\\0\end{pmatrix} \tag{77}$$

$$|11\rangle = \begin{pmatrix}0\\1\end{pmatrix} \otimes \begin{pmatrix}0\\1\end{pmatrix} = \begin{pmatrix}0\\0\\0\\1\end{pmatrix} \tag{78}$$

というものである。ここで $\otimes$ は Kronecker 積

$$A \otimes B \equiv \begin{pmatrix} a_{11}B & \dots & a_{1N}B \\ \vdots & \ddots & \vdots \\ a_{N1}B & \dots & a_{NN}B \end{pmatrix} \tag{79}$$

である。ユニタリー変換として $|10\rangle, |11\rangle$ だけに作用する

$$\Lambda(V) = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & a & b \\ 0 & 0 & c & d \end{pmatrix} \tag{80}$$

を考える。これは

$$\Lambda(V) = |0\rangle\langle 0| \otimes I + |1\rangle\langle 1| \otimes V, \quad |0\rangle\langle 0| = \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix}, \quad |1\rangle\langle 1| = \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix}, \quad V = \begin{pmatrix} a & b \\ c & d \end{pmatrix} \tag{81}$$

と書くことができる。このように、$|0\rangle, |1\rangle$ への射影演算子を用いて書かれるゲートを制御 V ゲートと呼ぶ。射影演算子が作用する bit を制御 bit, ユニタリー演算子が作用する bit を target bit と呼ぶ。

次に、$|00\rangle$ と $|11\rangle$ だけに作用するもの

$$\begin{pmatrix} a & 0 & 0 & b \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & b \\ c & 0 & 0 & d \end{pmatrix} \tag{82}$$

を考える。この場合も

1. $|00\rangle$ と $|10\rangle$ を swap する。
2. V を作用させる。
3. $|00\rangle$ と $|10\rangle$ を swap する。

によって、$\Lambda(V)$ を作用させる計算に帰着できる。$|00\rangle$ と $|10\rangle$ を swap は、「2 番目の bit が 0 ならば 1 番目の bit を反転させせよ」と読めば、制御演算で表せることが分かる。具体的には

$$\mathrm{SWAP} = X \otimes |0\rangle\langle 0| + I \otimes |1\rangle\langle 1| \tag{83}$$

である。この作用を具体的に計算すると、

$$\mathrm{SWAP}\,|00\rangle = |10\rangle \tag{84}$$
$$\mathrm{SWAP}\,|01\rangle = |01\rangle \tag{85}$$
$$\mathrm{SWAP}\,|10\rangle = |00\rangle \tag{86}$$
$$\mathrm{SWAP}\,|11\rangle = |11\rangle \tag{87}$$

となる。行列表示すると

$$\mathrm{SWAP} = \begin{pmatrix} 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \tag{88}$$

である。具体的に計算してみると

$$\mathrm{SWAP}\,\Lambda(V)\,\mathrm{SWAP} = \begin{pmatrix} 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & a & b \\ 0 & 0 & c & d \end{pmatrix} \begin{pmatrix} 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \tag{89}$$

$$= \begin{pmatrix} 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ a & 0 & 0 & b \\ c & 0 & 0 & d \end{pmatrix} \tag{90}$$

$$= \begin{pmatrix} a & 0 & 0 & b \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & b \\ c & 0 & 0 & d \end{pmatrix} \tag{91}$$

となって確かに所望のユニタリー変換が得られている。

このようにして、bit swap と制御ユニタリー変換をを組み合わせれば、任意の 2 level ユニタリー変換を構成することができる。

制御演算のうち、

$$\Lambda(X) = |0\rangle\langle 0| \otimes I + |1\rangle\langle 1| \otimes X, \tag{92}$$

を制御 NOT ゲート、あるいは CNOT ゲートと呼ぶ。実は、任意の制御ユニタリーゲートは、CNOT, H, T を用いて表すことができる。

4.4 3 bit 以上のユニタリー変換

3 bit 系で swap をやろうとすると、制御 bit が 2 つの NOT が必要になる。これを Toffoli ゲートと呼ぶ。Toffoli ゲートは CNOT と T ゲートに分解できる。また、Toffoli ゲートを組み合わせれば、制御 bit が 3 つ以上の NOT も作れる。こうして、CNOT, H, T の 3 つのゲートによって任意のユニタリー変換をいくらでも精度良く近似できることが分かった。この 3 つのゲートが、量子計算機における万能の組である。

5　基本的な量子アルゴリズム

5.1　量子計算機で実行できる操作のまとめ

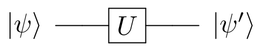
量子計算には重要な制約がある。

1. アルゴリズムはユニタリー演算として表されなければならない
2. 出力を得るために測定と呼ばれる操作を行わなければならない

各 qubit に対してユニタリー演算子を作用させることができる。例えば

$$U|\psi\rangle = |\psi'\rangle \tag{93}$$

といった操作が可能である。これを、



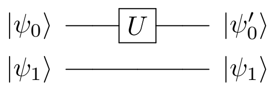
と表す。複数の qubit がある場合は、どの qubit に作用するのかを明示するために

$$U_j = I \otimes \cdots \otimes U \otimes \cdots \otimes I \tag{94}$$

という記法を用いる。例えば、

$$U_0|\psi_0\rangle|\psi_1\rangle = (U|\psi_0\rangle)|\psi_1\rangle = |\psi_0'\rangle|\psi_1\rangle \tag{95}$$

である。これは以下のように表す。



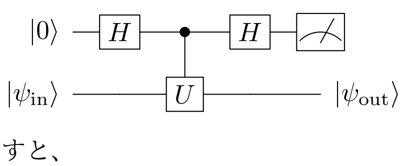
上記に加えて、制御ユニタリー演算というものを作用させることができる。j 番の qubit を制御系とする k 番の qubit に対する制御ユニタリー演算とは、

$$\Lambda_{jk}(U) = \cdots \otimes \underbrace{|0\rangle\langle 0|\, I}_{j} \otimes \cdots \otimes \underbrace{|1\rangle\langle 1|\, I}_{k} \otimes \cdots \tag{96}$$

である。$\cdots$ で省略したところにはすべて I が入る。特に、$U = X$ の場合を CNOT ゲートと呼ぶ。

5.2　アダマールテスト

U をユニタリー演算子とする。以下のゲート



を考える。control U gate を式で表すと、

$$|0\rangle\langle 0| \otimes I + |1\rangle\langle 1| \otimes U \tag{97}$$

であることに注意して、この回路を式で書くと、測定の直前の状態は

$$\begin{aligned}
&(H\otimes I)(|0\rangle\langle 0|\otimes I+|1\rangle\langle 1|\otimes U)(H\otimes I)|0\rangle|\psi_{\mathrm{in}}\rangle && (98)\\
=&(H\otimes I)(|0\rangle\langle 0|\otimes I+|1\rangle\langle 1|\otimes U)\frac{|0\rangle|\psi_{\mathrm{in}}\rangle+|1\rangle|\psi_{\mathrm{in}}\rangle}{\sqrt{2}} && (99)\\
=&(H\otimes I)\left(\frac{|0\rangle|\psi_{\mathrm{in}}\rangle}{\sqrt{2}}+\frac{|1\rangle U|\psi_{\mathrm{in}}\rangle}{\sqrt{2}}\right) && (100)\\
=&\frac{1}{\sqrt{2}}(H\otimes I)\left(|0\rangle|\psi_{\mathrm{in}}\rangle+|1\rangle U|\psi_{\mathrm{in}}\rangle\right) && (101)\\
=&\frac{1}{\sqrt{2}}\left(\frac{|0\rangle+|1\rangle}{\sqrt{2}}|\psi_{\mathrm{in}}\rangle+\frac{|0\rangle-|1\rangle}{\sqrt{2}}U|\psi_{\mathrm{in}}\rangle\right) && (102)\\
=&\frac{|0\rangle+|1\rangle}{2}|\psi_{\mathrm{in}}\rangle+\frac{|0\rangle-|1\rangle}{2}U|\psi_{\mathrm{in}}\rangle && (103)\\
=&|0\rangle\frac{I+U}{2}|\psi_{\mathrm{in}}\rangle+|1\rangle\frac{I-U}{2}|\psi_{\mathrm{in}}\rangle && (104)
\end{aligned}$$

となる。第一の qubit が $|0\rangle$ である確率は

$$\begin{aligned}
p_0&=\left|(|0\rangle\langle 0|\otimes I)\left(|0\rangle\frac{I+U}{2}|\psi_{\mathrm{in}}\rangle+|1\rangle\frac{I-U}{2}|\psi_{\mathrm{in}}\rangle\right)\right|^2 && (105)\\
&=\left||0\rangle\frac{I+U}{2}|\psi_{\mathrm{in}}\rangle\right|^2 && (106)\\
&=\left(\langle\psi_{\mathrm{in}}|\frac{I+U^\dagger}{2}\langle 0|\right)\left(|0\rangle\frac{I+U}{2}|\psi_{\mathrm{in}}\rangle\right) && (107)\\
&=\langle\psi_{\mathrm{in}}|\frac{I+U+U^\dagger+U^\dagger U}{4}|\psi_{\mathrm{in}}\rangle && (108)\\
&=\langle\psi_{\mathrm{in}}|\frac{2I+U+U^\dagger}{4}|\psi_{\mathrm{in}}\rangle && (109)\\
&=\frac{1+\mathrm{Re}\langle\psi_{\mathrm{in}}|U|\psi_{\mathrm{in}}\rangle}{2} && (110)
\end{aligned}$$

となり、第一の qubit が $|1\rangle$ である確率は

$$\begin{aligned}
p_1&=\left|(|1\rangle\langle 1|\otimes I)\left(|0\rangle\frac{I+U}{2}|\psi_{\mathrm{in}}\rangle+|1\rangle\frac{I-U}{2}|\psi_{\mathrm{in}}\rangle\right)\right|^2 && (111)\\
&=\left||1\rangle\frac{I-U}{2}|\psi_{\mathrm{in}}\rangle\right|^2 && (112)\\
&=\left(\langle\psi_{\mathrm{in}}|\frac{I-U^\dagger}{2}\langle 1|\right)\left(|1\rangle\frac{I-U}{2}|\psi_{\mathrm{in}}\rangle\right) && (113)\\
&=\langle\psi_{\mathrm{in}}|\frac{I-U-U^\dagger+U^\dagger U}{4}|\psi_{\mathrm{in}}\rangle && (114)\\
&=\langle\psi_{\mathrm{in}}|\frac{2I-U-U^\dagger}{4}|\psi_{\mathrm{in}}\rangle && (115)\\
&=\frac{1-\mathrm{Re}\langle\psi_{\mathrm{in}}|U|\psi_{\mathrm{in}}\rangle}{2} && (116)
\end{aligned}$$

となる。従って、この回路では演算子 U の $\langle\psi_{\mathrm{in}}|$ における期待値を推定することができる。

測定の結果、第一番目の qubit が $|0\rangle,|1\rangle$ だった場合、残りの状態はそれぞれ、

$$|\psi_{\mathrm{out}}\rangle=|\psi_0\rangle=\frac{I+U}{2}|\psi_{\mathrm{in}}\rangle,\quad|\psi_{\mathrm{out}}\rangle=|\psi_1\rangle=\frac{I-U}{2}|\psi_{\mathrm{in}}\rangle \tag{117}$$

となる。

5.3 量子フーリエ変換

x_j を 2^n 成分ベクトルとする。これは規格化 $\sum_{j=0}^{2^n-1}|x_j|^2=1$ されているとする。この離散フーリエ変換

$$y_k=\frac{1}{\sqrt{2^n}}\sum_{j=0}^{2^n-1}x_j e^{i\frac{2\pi jk}{2^n}} \tag{118}$$

を量子回路を用いて計算する方法について述べる。

整数 j に対してその 2 進数表記をラベルに持つような量子状態を考え、次のように書く。

$$|(j)_2\rangle=|i_1 i_2\cdots\rangle \tag{119}$$

例えば

$$|(6)_2\rangle=|110\rangle=|1\rangle\otimes|1\rangle\otimes|0\rangle=\begin{pmatrix}0\\0\\0\\0\\0\\0\\1\\0\end{pmatrix} \tag{120}$$

である。この約束のもと、次のような状態を考える

$$|x\rangle\rangle\equiv\sum_{j=0}^{2^n-1}x_j\,|(j)_2\rangle\,,\quad |y\rangle\rangle\equiv\sum_{j=0}^{2^n-1}y_j\,|(j)_2\rangle \tag{121}$$

$|y\rangle\rangle$ を x_j で表すと

$$|y\rangle\rangle=\sum_{k=0}^{2^n-1}y_k\,|(k)_2\rangle \tag{122}$$

$$=\frac{1}{\sqrt{2^n}}\sum_{j=0}^{2^n-1}\sum_{k=0}^{2^n-1}x_j e^{i\frac{2\pi jk}{2^n}}\,|(k)_2\rangle \tag{123}$$

$$=\sum_{j=0}^{2^n-1}x_j\left(\frac{1}{\sqrt{2^n}}\sum_{k=0}^{2^n-1}e^{i\frac{2\pi jk}{2^n}}\,|(k)_2\rangle\right) \tag{124}$$

となる。もし、あるユニタリー変換で、

$$U\,|(j)_2\rangle=\frac{1}{\sqrt{2^n}}\sum_{k=0}^{2^n-1}e^{i\frac{2\pi jk}{2^n}}\,|(k)_2\rangle \tag{125}$$

となるようなものがあったとすると、

$$U\,|x\rangle\rangle=|y\rangle\rangle \tag{126}$$

となる。$|y\rangle\rangle$ の係数を読み取ることで、フーリエ変換の結果を知ることができる。

以下で、そのような U を具体的に構成する。ビット数は n で固定する。

$$\begin{aligned}
\sum_{k=0}^{2^n-1} e^{i\frac{2\pi jk}{2^n}} \left|(k)_2\right\rangle &= \sum_{k_1=0,1} \cdots \sum_{k_n=0,1} e^{i\frac{2\pi j(k_1 2^{n-1}+\cdots+k_0 2^0)}{2^n}} \left|k_1 \cdots k_n\right\rangle && (127)\\
&= \sum_{k_1=0,1} \cdots \sum_{k_n=0,1} e^{i2\pi j(k_1 2^{-1}+\cdots+k_0 2^{-n})} \left|k_1 \cdots k_n\right\rangle && (128)\\
&= \left(\sum_{k_1=0,1} e^{i2\pi jk_1 2^{-1}} \left|k_1\right\rangle\right) \otimes \cdots \otimes \left(\sum_{k_n=0,1} e^{i2\pi jk_n 2^{-n}} \left|k_n\right\rangle\right) && (129)\\
&= \left(|0\rangle + e^{i2\pi j2^{-1}} |1\rangle\right) \otimes \cdots \otimes \left(|0\rangle + e^{i2\pi j2^{-n}} |1\rangle\right) && (130)
\end{aligned}$$

ここで、$j2^{-l}$ という因子の 2 進数表記について考える。

$$(j)_2 = j_1 j_2 \cdots j_n \tag{131}$$

とすると、

$$j = j_1 2^{n-1} + j_2 2^{n-2} + \cdots + j_n 2^0 \tag{132}$$

であるから

$$j2^{-l} = j_1 2^{n-l-1} + j_2 2^{n-l-2} + \cdots + j_n 2^{-l} \tag{133}$$

である。よって、これを 2 進数表記すると

$$\begin{aligned}
(j2^{-1})_2 &= (\text{整数部分}).j_n && (134)\\
(j2^{-2})_2 &= (\text{整数部分}).j_{n-1}j_n && (135)\\
&\vdots && (136)\\
(j2^{-l})_2 &= (\text{整数部分}).j_{n-l+1}\cdots j_{n-1}j_n && (137)\\
&\vdots && (138)\\
(j2^{-n})_2 &= (\text{整数部分}).j_1\cdots j_{n-1}j_n && (139)\\
& && (140)
\end{aligned}$$

となる。また、一般に

$$\begin{aligned}
e^{i2\pi j_1\cdots j_{l-1}.j_l\cdots j_n} &= e^{i2\pi\left(\cdots+j_{l-2}2^1+j_{l-1}+\frac{j_l}{2^1}\cdots+\frac{j_n}{2^{n-l+1}}\right)} && (141)\\
&= \cdots e^{i2\pi j_{l-2}2^1} e^{i2\pi j_{l-1}} e^{i2\pi\frac{j_l}{2^1}} \cdots e^{i2\pi\frac{j_n}{2^{n-l+1}}} && (142)\\
&= e^{i2\pi\frac{j_l}{2^1}} \cdots e^{i2\pi\frac{j_n}{2^{n-l+1}}} && (143)\\
&= e^{i2\pi\left(\frac{j_l}{2^1}\cdots+\frac{j_n}{2^{n-l+1}}\right)} && (144)\\
&= e^{i2\pi 0.j_l\cdots j_n} && (145)
\end{aligned}$$

のように整数部分は効いてこないことに注意すると、

$$\begin{aligned}
\sum_{k=0}^{2^n-1} e^{i\frac{2\pi jk}{2^n}} \left|(k)_2\right\rangle &= \left(|0\rangle + e^{i2\pi j2^{-1}} |1\rangle\right) \otimes \cdots \otimes \left(|0\rangle + e^{i2\pi j2^{-n}} |1\rangle\right) && (146)\\
&= \left(|0\rangle + e^{i2\pi 0.j_n} |1\rangle\right) \otimes \cdots \otimes \left(|0\rangle + e^{i2\pi 0.j_1\cdots j_n} |1\rangle\right) && (147)
\end{aligned}$$

を得る。よって、求めるべきユニタリー変換 U とは

$$U\left|(j)_2\right\rangle = \frac{1}{\sqrt{2^n}}\left(|0\rangle + e^{i2\pi 0.j_n}|1\rangle\right)\otimes\cdots\otimes\left(|0\rangle + e^{i2\pi 0.j_1\cdots j_n}|1\rangle\right) \tag{148}$$

となるようなものである。

まず、Hadamard ゲートが状態に対して

$$H|j\rangle = \frac{|0\rangle + (-1)^j|1\rangle}{\sqrt{2}} \tag{149}$$

と作用することを思い出す。ここで、2 進小数を導入すると

$$e^{i2\pi 0.0} = 1, \quad e^{i2\pi 0.1} = e^{\frac{i2\pi}{2}} = -1 \tag{150}$$

であるから、

$$H|j\rangle = \frac{|0\rangle + e^{i2\pi 0.j}|1\rangle}{\sqrt{2}} \tag{151}$$

と表すことができる。このことを用い、まず第一番目の bit に Hadamard ゲートを作用させると

$$(H\otimes I\otimes\cdots)\,|j_1j_2\cdots j_n\rangle = \left(\frac{|0\rangle + e^{i2\pi 0.j_1}|1\rangle}{\sqrt{2}}\right)|j_2\cdots j_n\rangle \tag{152}$$

となる。次に、2 番目の bit を制御ゲートとする位相ゲート R_2 を一番目のゲートに作用させる。3 番目以降の bit は関与しないから、1,2 番目の bit だけに注目して計算してみる。

$$(I\otimes|0\rangle\langle 0| + R_2\otimes|1\rangle\langle 1|)\left(\frac{|0\rangle + e^{i2\pi 0.j_1}|1\rangle}{\sqrt{2}}\right)|0\rangle = \frac{|00\rangle + e^{i2\pi 0.j_1}|10\rangle}{\sqrt{2}} = \frac{|00\rangle + e^{i2\pi 0.j_1 0}|10\rangle}{\sqrt{2}} \tag{153}$$

$$(I\otimes|0\rangle\langle 0| + R_2\otimes|1\rangle\langle 1|)\left(\frac{|0\rangle + e^{i2\pi 0.j_1}|1\rangle}{\sqrt{2}}\right)|1\rangle = \left(\frac{|0\rangle + e^{i\frac{2\pi}{2^2}}e^{i2\pi 0.j_1}|1\rangle}{\sqrt{2}}\right)|1\rangle = \frac{|01\rangle + e^{i2\pi 0.j_1 1}|11\rangle}{\sqrt{2}} \tag{154}$$

この結果をまとめると、

$$(I\otimes|0\rangle\langle 0| + R_2\otimes|1\rangle\langle 1|)\left(\frac{|0\rangle + e^{i2\pi 0.j_1}|1\rangle}{\sqrt{2}}\right)|j_2\rangle = \frac{|0\rangle + e^{i2\pi 0.j_1j_2}|1\rangle}{\sqrt{2}}|j_2\rangle \tag{155}$$

となる。まったく同様の計算により、

$$C_{n,1}(R_n)\cdots C_{3,1}(R_3)C_{2,1}(R_2)H_1\,|j_1j_2j_3\cdots j_n\rangle = \frac{|0\rangle + e^{i2\pi 0.j_1j_2j_3\cdots j_n}|1\rangle}{\sqrt{2}}|j_2j_3\cdots j_n\rangle \tag{156}$$

となる。以下、残った $|j_2j_3\cdots j_n\rangle$ の部分に同様な操作を施す。

$$C_{n-1,2}(R_n)\cdots C_{3,2}(R_4)C_{2,2}(R_3)H_2\,|\bullet j_2j_3\cdots j_n\rangle = \frac{|0\rangle + e^{i2\pi 0.j_2j_3\cdots j_n}|1\rangle}{\sqrt{2}}|j_3\cdots j_n\rangle \tag{157}$$

$$C_{n-1,3}(R_n)\cdots C_{3,3}(R_5)C_{2,3}(R_4)H_3\,|\bullet\bullet j_3\cdots j_n\rangle = \frac{|0\rangle + e^{i2\pi 0.j_3\cdots j_n}|1\rangle}{\sqrt{2}}|j_4\cdots j_n\rangle \tag{158}$$

$$\vdots \tag{159}$$

これらをすべて合わせると、

$$|j_1j_2\cdots j_n\rangle \to \frac{1}{\sqrt{2^n}}\left(|0\rangle + e^{i2\pi 0.j_1\cdots j_n}|1\rangle\right)\otimes\cdots\otimes\left(|0\rangle + e^{i2\pi 0.j_n}|1\rangle\right) \tag{160}$$

を得る。最後に SWAP ゲートで状態の入れ替えを行えば完了である。

5.4 位相推定アルゴリズム

ユニタリー演算子 U が固有値 $e^{i\lambda_l}$ をもち、対応する固有状態が $|\psi_l\rangle$ であるとする。固有値の位相は $0 \leq \lambda_l \leq 2\pi$ として一般性を失わないから、$\lambda_l/2\pi$ は次のように 2 進展開できる:

$$\frac{\lambda_l}{2\pi} = \frac{j_1^{(l)}}{2^1} + \cdots + \frac{j_n^{(l)}}{2^n} \tag{161}$$

ただし、展開が無限次になる場合は、n 桁で打ち切って近似しているものと考える。これを次のように表す。

$$\lambda_l = (2\pi)0.j_1^{(l)} \cdots j_n^{(l)} \tag{162}$$

一般の量子状態 $|\psi\rangle$ を U の固有状態で展開したときの係数を c_l と表すことができる。

$$|\psi\rangle = \sum_l c_l |\psi_l\rangle \tag{163}$$

このとき、$|\psi\rangle$ と n 桁の補助量子ビット $|00\ldots0\rangle$ を

$$V|00\ldots0\rangle|\psi\rangle = \sum_l c_l |j_1^{(l)} \cdots j_n^{(l)}\rangle |\psi_l\rangle \tag{164}$$

のように変換するアルゴリズムを量子位相推定と呼ぶ。例えば、補助ビットが $|0\cdots0\rangle$ となる確率は

$$\left| |0\cdots0\rangle\langle0\cdots0| \otimes I \sum_l c_l |j_1^{(l)} \cdots j_n^{(l)}\rangle |\psi_l\rangle \right|^2 \tag{165}$$

$$= \left| \sum_l c_l |0\cdots0\rangle\langle0\cdots0| |j_1^{(l)} \cdots j_n^{(l)}\rangle |\psi_l\rangle \right|^2 \tag{166}$$

$$= \sum_{ll'} c_l c_{l'}^* \langle\psi_{l'}| \langle j_1^{(l')} \cdots j_n^{(l')}| |0\cdots0\rangle\langle0\cdots0| |0\cdots0\rangle\langle0\cdots0| |j_1^{(l)} \cdots j_n^{(l)}\rangle |\psi_l\rangle \tag{167}$$

$$= \sum_{ll'} c_l c_{l'}^* \delta_{ll'} \langle j_1^{(l')} \cdots j_n^{(l')}| |0\cdots0\rangle\langle0\cdots0| |j_1^{(l)} \cdots j_n^{(l)}\rangle \tag{168}$$

$$= \sum_l |c_l|^2 \langle j_1^{(l)} \cdots j_n^{(l)}| |0\cdots0\rangle\langle0\cdots0| |j_1^{(l)} \cdots j_n^{(l)}\rangle \tag{169}$$

$$= \sum_l |c_l|^2 \left| \langle0\cdots0| |j_1^{(l)} \cdots j_n^{(l)}\rangle \right|^2 \tag{170}$$

まず準備として、ひとつの補助ビットと U の固有状態のテンソル積状態を考え、それに Hadamard ゲートと制御 U^{2^k} ゲートを作用させる。

$$C_1(U_2^{2^k})H_1|0\rangle|\psi_l\rangle = C_1(U_2^{2^k})\frac{|0\rangle + |1\rangle}{\sqrt{2}}|\psi_l\rangle \tag{171}$$

$$= \frac{|0\rangle}{\sqrt{2}}|\psi_l\rangle + \frac{|1\rangle}{\sqrt{2}}U^{2^k}|\psi_l\rangle \tag{172}$$

$$= \frac{|0\rangle + e^{2^k i\lambda_l}|1\rangle}{\sqrt{2}}|\psi_l\rangle \tag{173}$$

ここで、

$$2^0\lambda_l = (2\pi)2^0 0.j_1^{(l)}\cdots j_n^{(l)} = (2\pi)0.j_2^{(l)}\cdots j_n^{(l)} \tag{174}$$
$$2^1\lambda_l = (2\pi)2^1 0.j_1^{(l)}\cdots j_n^{(l)} = (2\pi)j_1^{(l)}.j_2^{(l)}\cdots j_n^{(l)} \tag{175}$$
$$2^2\lambda_l = (2\pi)2^2 0.j_1^{(l)}\cdots j_n^{(l)} = (2\pi)j_1^{(l)}j_2^{(l)}.j_3^{(l)}\cdots j_n^{(l)} \tag{176}$$
$$\vdots \tag{177}$$
$$2^k\lambda_l = (2\pi)2^k 0.j_1^{(l)}\cdots j_n^{(l)} = (2\pi)j_1^{(l)}\cdots j_k^{(l)}.j_{k+1}^{(l)}\cdots j_n^{(l)} \tag{178}$$

であるから、

$$C_1(U_2^{2^k})H_1|0\rangle|\psi_l\rangle = \frac{|0\rangle + e^{2^k i\lambda_l}|1\rangle}{\sqrt{2}}|\psi_l\rangle \tag{179}$$
$$= \frac{|0\rangle + e^{(2\pi i)j_1^{(l)}\cdots j_k^{(l)}.j_{k+1}^{(l)}\cdots j_n^{(l)}}|1\rangle}{\sqrt{2}}|\psi_l\rangle \tag{180}$$
$$= \frac{|0\rangle + e^{(2\pi i)0.j_{k+1}^{(l)}\cdots j_n^{(l)}}|1\rangle}{\sqrt{2}}|\psi_l\rangle \tag{181}$$

を得る。この結果を用いると、

$$\prod_{k=1}^{n} C_k(U_2^{2^{k-1}})H_k|00\cdots 0\rangle|\psi_l\rangle \tag{182}$$
$$=\frac{1}{\sqrt{2^n}}\left(|0\rangle + e^{(2\pi i)0.j_1^{(l)}\cdots j_n^{(l)}}|1\rangle\right)\left(|0\rangle + e^{(2\pi i)0.j_2^{(l)}\cdots j_n^{(l)}}|1\rangle\right)\cdots\left(|0\rangle + e^{(2\pi i)0.j_n^{(l)}}|1\rangle\right)|\psi_l\rangle \tag{183}$$

となるが、これは $|j_1j_2\cdots j_n\rangle$ に量子フーリエ変換を施してできる状態に他ならない。従って、

$$\mathrm{QFT}^\dagger\prod_{k=1}^{n} C_k(U_2^{2^{k-1}})H_k|00\cdots 0\rangle|\psi_l\rangle = |j_1^{(l)}j_2^{(l)}\cdots j_n^{(l)}\rangle|\psi_l\rangle \tag{184}$$

である。よって、$V = \mathrm{QFT}^\dagger\prod_{k=1}^{n} C_k(U_2^{2^{k-1}})H_k$ とすれば良いことが分かった。

5.5 Shor の素因数分解アルゴリズム

正の N を整数とし、これを素因数分解したいものとする。N と互いに素な数 x を用意し、

$$x^r = 1 \mod N \tag{185}$$

とおく。これを満たす最小の r を x の位数と呼ぶ。大概の場合、位数は偶数になることが知られているらしいので、以下それを仮定する。このとき、$x^r - 1$ を因数分解することができて

$$(x^{r/2}-1)(x^{r/2}+1) = 0 \mod N \tag{186}$$

である。これが成り立つのは

- $x^{r/2} \pm 1 = 0 \mod N$ である。
- $(x^{r/2}-1)$, $(x^{r/2}+1)$ が N と非自明な公約数を持つ。

のどちらかの場合であるが、実は後者になる場合が多いらしいので、以下それを仮定する。するとユークリッドの互除法により、$(x^{r/2}-1)$, $(x^{r/2}+1)$ が N と非自明な公約数を計算することができ、N の因数がひとつ判明する。これを繰り返すことで素因数分解が完了する。

例) うまくいく場合。

1. $N=57$ とする。
2. $x=5$ を取る。
3. $r=18$ となる。
4. $x^{r/2}-1=5^9-1=1953124$, $x^{r/2}+1=5^9+1=1953126$ である。これらは 57 では割り切れない。
5. $\gcd(5^9-1,57)=19$, $\gcd(5^9+1,57)=3$ である。

失敗する場合。

1. $N=57$ とする。
2. $x=2$ を取る。
3. $r=18$ となる。
4. $2^9-1=511$, $2^9+1=513$ である。$513/57=9$ である。
5. $\gcd(2^9-1,57)=1$, $\gcd(2^9+1,57)=57$ である。

さて、この計算で最もしんどいのは r を求める部分であるが、これを量子計算機にやらせることができる。まず、与えられた N,x に対して、ユニタリー演算子

$$U_x=\sum_{y=0}^{N-1}|xy \mod N\rangle\langle y| \tag{187}$$

を考える。この固有値と固有ベクトルは

$$U_x|u_s\rangle=e^{2\pi i(s/r)}|u_s\rangle, \quad s=0,\cdots,r-1 \tag{188}$$

$$|u_s\rangle=\frac{1}{\sqrt{r}}\sum_{k=0}^{r-1}e^{-2\pi i(s/r)k}|x^k \mod N\rangle \tag{189}$$

である。実際、

$$U_x|u_s\rangle = \frac{1}{\sqrt{r}}\sum_{y=0}^{N-1}\sum_{k=0}^{r-1} e^{-2\pi i(s/r)k}|xy \mod N\rangle\langle y|x^k \mod N\rangle \tag{190}$$

$$= \frac{1}{\sqrt{r}}\sum_{k=0}^{r-1} e^{-2\pi i(s/r)k}|x^{k+1} \mod N\rangle \tag{191}$$

$$= \frac{1}{\sqrt{r}}\sum_{k=0}^{r-2} e^{-2\pi i(s/r)k}|x^{k+1} \mod N\rangle + \frac{1}{\sqrt{r}}e^{-2\pi i(s/r)(r-1)}|x^r \mod N\rangle \tag{192}$$

$$= \frac{1}{\sqrt{r}}\sum_{k=0}^{r-2} e^{-2\pi i(s/r)k}|x^{k+1} \mod N\rangle + \frac{1}{\sqrt{r}}e^{-2\pi i(s/r)(r-1)}|1 \mod N\rangle \tag{193}$$

$$= \frac{1}{\sqrt{r}}\sum_{k=1}^{r-1} e^{-2\pi i(s/r)(k-1)}|x^k \mod N\rangle + \frac{1}{\sqrt{r}}e^{-2\pi i(s/r)(r-1)}|x^0 \mod N\rangle \tag{194}$$

$$= e^{2\pi i(s/r)}\frac{1}{\sqrt{r}}\left(\sum_{k=1}^{r-1} e^{-2\pi i(s/r)k}|x^k \mod N\rangle + e^{-2\pi i(s/r)r}|x^0 \mod N\rangle\right) \tag{195}$$

$$= e^{2\pi i(s/r)}\frac{1}{\sqrt{r}}\left(\sum_{k=1}^{r-1} e^{-2\pi i(s/r)k}|x^k \mod N\rangle + |x^0 \mod N\rangle\right) \tag{196}$$

$$= e^{2\pi i(s/r)}\frac{1}{\sqrt{r}}\sum_{k=0}^{r-1} e^{-2\pi i(s/r)k}|x^k \mod N\rangle \tag{197}$$

$$= e^{2\pi i(s/r)}|u_s\rangle \tag{198}$$

となっている。従って、U_x を量子位相推定することにより、s/r を得ることができる。

例えば $N=6, x=5$ の場合、$r=2$ である。U の行列表示は

$$U_5 = \sum_{y=0}^{5}|5y \mod 6\rangle\langle y| \tag{199}$$

$$= |0 \mod 6\rangle\langle 0| + |5 \mod 6\rangle\langle 1| + |10 \mod 6\rangle\langle 2| + |15 \mod 6\rangle\langle 3| + |20 \mod 6\rangle\langle 4| + |25 \mod 6\rangle\langle 5| \tag{200}$$

$$= |0\rangle\langle 0| + |5\rangle\langle 1| + |4\rangle\langle 2| + |3\rangle\langle 3| + |2\rangle\langle 4| + |1\rangle\langle 5| \tag{201}$$

$$= \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \end{pmatrix} \tag{202}$$

である。固有値は $e^{2\pi i\times 0}=1$, $e^{2\pi i(1/2)}=-1$ である。（縮退がある）状態を測定して後者が得られれば、$r=2$ が得られる。

6　誤り訂正

6.1　線形符号による古典誤り訂正

最も素朴な誤り訂正の方法は多数決である。例えば、0 というデータがあったとき、これを 000 という風に冗長化したデータを作っておけば、何らかの要因によってビットが部分的に反転してしまい 010 になったとしても、反転前のデータは 000 であろうと推定できる。

k ビットの情報 v（成分が 0 or 1 の k 成分ベクトル）があったとし、これを $n = dk$ ビットのベクトルに冗長化することを考える。冗長化後のベクトルを v', v から v' への変換を G とする。

$$v' = Gv \tag{203}$$

G は $n \times k$ 行列である。この操作を符号化、G を生成行列と呼ぶ。G として冒頭の多数決方式を採用したものを線形符号と呼ぶ。例えば、

$$v = \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix}, \quad G = \begin{pmatrix} 1 & 0 & 0 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \end{pmatrix}, \tag{204}$$

と取れば、

$$v' = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \\ 1 \\ 1 \\ 0 \\ 0 \\ 0 \end{pmatrix} \tag{205}$$

となる。

次に、エラーが起きたかどうかを判定する方法を考える。n ビットのベクトルのうち、Gv の形に表されるもの全体の集合を W とする。W の元を符号語と呼ぶ。例えば $(k = 2,\, d = 3)$

$$v = \begin{pmatrix} 1 \\ 1 \\ 1 \\ 0 \\ 0 \\ 0 \end{pmatrix} \tag{206}$$

は符号語だが、

$$v = \begin{pmatrix} 1 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \end{pmatrix} \tag{207}$$

ではない。

各列ベクトルが独立で、$H_c G \equiv 0 \pmod 2$ を満たすような $(n-k) \times n$ 行列 H_c があったとする。これを検査行列と呼ぶ。任意の符号語 w について

$$H_c w = H_c G v = 0 \tag{208}$$

が成り立つ。また逆に、$H_c w = 0$ ならば、w は符号語であることも示せるらしい。このようにして、符号語の 2 通りの特徴付けを得た。

- G を用いた見方: ベクトルが適切に水増しされている。
- H_c を用いた見方: ブロック内の隣接するビットが等しい。

$w \in W$ のときに限りゼロになるようなベクトル $s = H_c w$ を w のシンドローム、その成分をシンドローム値と呼ぶ。ひとつでも 0 でないシンドローム値があれば、w にはエラーが起きていることになる。シンドローム値がすべて 0 のときは、w にはエラーがないか、エラー自身が $H_c e = 0$ を満たしているかのどちらかである。

（例 $k = 3, n = 9$）線形符号 G に対する検査行列は以下を取ればよい。

$$H_c = \begin{pmatrix} 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \end{pmatrix} \tag{209}$$

実際

$$\begin{pmatrix} 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} 2 & 0 & 0 \\ 2 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 2 \\ 0 & 0 & 2 \end{pmatrix} \equiv \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} \tag{210}$$

となり OK である。符号語 w に対して $H_c w$ を計算すると、

$$H_c w = \begin{pmatrix} 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \end{pmatrix} \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \\ 1 \\ 1 \\ 0 \\ 0 \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 2 \\ 2 \\ 0 \\ 0 \end{pmatrix} \equiv \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix} \tag{211}$$

となる。もしエラーが混入していると、

$$H_c w = \begin{pmatrix} 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \end{pmatrix} \left(\begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \\ 1 \\ 1 \\ 0 \\ 0 \\ 0 \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix} \right) = \left(\begin{pmatrix} 0 \\ 0 \\ 2 \\ 2 \\ 0 \\ 0 \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \end{pmatrix} \right) \equiv \begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \end{pmatrix} \tag{212}$$

となる。

6.2 量子誤り訂正

現在知られている量子誤り訂正は、線形符号を量子計算版とでもいうべきものである。具体例から考える。1 ビットの情報を 3 ビットに冗長化することを考える。ここでは、符号化前の量子状態もあらかじめ 3bit で用意しておき、情報は 1bit 目に格納しておくことにする。よって符号化 G として

$$|000\rangle \to |000\rangle \tag{213}$$
$$|100\rangle \to |111\rangle \tag{214}$$

となるようなものを用意できれば良い。これは CNOT ゲート

$$\Lambda_{1,2}(X) = P_0 \otimes I \otimes I + P_1 \otimes X \otimes I \tag{215}$$
$$\Lambda_{1,3}(X) = P_0 \otimes I \otimes I + P_1 \otimes I \otimes X \tag{216}$$

で実現できる。実際、

$$\Lambda_{1,2}(X)\Lambda_{1,3}(X) |000\rangle = |000\rangle \tag{217}$$
$$\Lambda_{1,2}(X)\Lambda_{1,3}(X) |100\rangle = |111\rangle \tag{218}$$

となる。

次にシンドロームを調べる方法を考える。

$$M_0^{(1)} = \frac{I + Z_1 Z_2}{2}, \quad M_1^{(1)} = \frac{I - Z_1 Z_2}{2} \tag{219}$$

という 2 つの演算子を考えると、これらは POVM をなし、

$$M_0^{(1)} |000\rangle = \frac{1+1}{2} |000\rangle = |000\rangle \tag{220}$$
$$M_0^{(1)} |100\rangle = \frac{1-1}{2} |100\rangle = 0 \tag{221}$$
$$M_0^{(1)} |010\rangle = \frac{1-1}{2} |010\rangle = 0 \tag{222}$$
$$M_0^{(1)} |110\rangle = \frac{1+1}{2} |110\rangle = |110\rangle \tag{223}$$

$$M_1^{(1)} |000\rangle = \frac{1-1}{2} |000\rangle = 0 \tag{224}$$
$$M_1^{(1)} |100\rangle = \frac{1+1}{2} |100\rangle = |100\rangle \tag{225}$$
$$M_1^{(1)} |010\rangle = \frac{1+1}{2} |010\rangle = |010\rangle \tag{226}$$
$$M_1^{(1)} |110\rangle = \frac{1-1}{2} |110\rangle = 0 \tag{227}$$

より、

$$\mathrm{tr}(|ij0\rangle \langle ij0| M_0^1) = \delta_{ij}, \tag{228}$$
$$\mathrm{tr}(|ij0\rangle \langle ij0| M_1^1) = 1 - \delta_{ij} \tag{229}$$

となるから、M_1^1 の期待値が、1,2 番のビットに関するシンドローム値を与えることが分かる。全く同様にして、

$$M_0^{(2)} = \frac{I + Z_2 Z_3}{2}, \quad M_1^{(2)} = \frac{I - Z_2 Z_3}{2} \tag{230}$$

も測定すれば、すべてのシンドローム値が得られる。古典的な誤り訂正の場合、シンドローム値は 0 か 1 であったが、量子計算の場合はそれ以外の中途半端な値を取りうる。例えば、$|\psi\rangle = (|000\rangle + |010\rangle)/\sqrt{2}$ のとき、

$$M_0^{(1)} |\psi\rangle \frac{1}{\sqrt{2}}(|000\rangle + |010\rangle) = \frac{|000\rangle}{\sqrt{2}} \tag{231}$$

より、

$$\mathrm{tr}(|\psi\rangle \langle\psi| M_0^{(1)}) = \frac{1}{2} \tag{232}$$

となる。

以下では簡単のため、このような位相を変化させるエラーはなく、ビット反転を引き起こすエラーのみを考える。そのようなエラーを引き起こす操作は

$$E = X_1^{e_1} X_2^{e_1} X_3^{e_1} \tag{233}$$

と書ける。ただし、$e_1, e_2, e_3 = 0, 1$ である。仮に、$|000\rangle$ がオリジナルのデータで、これに、$e_1 = 1, e_2 = e_3 = 0$ のエラーが乗ったとすると、

$$E |000\rangle = X_1 |000\rangle = |100\rangle \tag{234}$$

となる。この状態のシンドローム値を完全に調べると、1 番目のビットに反転があることが分かるから、状態に再度 X_1 を作用させればもとの状態を復元することができる。

$$R\left|100\right\rangle = X_1\left|100\right\rangle = \left|000\right\rangle \tag{235}$$

これでエラー訂正が完了した。

6.3 スタビライザー符号

6.4 surface code

7 参考文献

- 量子コンピュータの基礎と物理の接点は藤井さんによる講義ノート。
- Quantum Computation and Quantum Information は Nielsen, Chuang による古典的な教科書。10 版が無料公開されている。
- Quantum Native Dojo は株式会社 QunaSys が運営しているサイト。
- 古典論理回路については、例えば、「論理回路　高木直史　昭晃堂」