

測定型量子計算

高波海斗

2022年4月29日

1 イントロ

量子力学の理論の発展によって、古典力学と異なる数多くの性質が自然界には存在することが明らかになりました。例えば古典力学では、ある時刻での速度と位置を指定したボールは、その後の運動を全て運動方程式により決定することができます。一方、量子力学に従う粒子は、その粒子の今の位置すら、一般的には決定的ではありません。このように事象が確率に左右されることは、自然を物理的に捉える上では一見面倒であるように思われます。しかしながら、量子情報ではこの原理をむしろ積極的に活用し、新しいアルゴリズムや通信プロトコルを発明することを考えます。この量子力学特有の不思議な性質を使った計算が、量子計算です。例として、多項式時間で素因数分解を行う Shor のアルゴリズムなどが有名です。

量子計算の方法としてよく知られているのが、ゲートと呼ばれるユニタリー変換を利用するものです(表1) [8]。まず、最初に適当な量子ビットを初期状態として用意し、そこに次々とユニタリー変換を施していきます。そして一番最後に測定を行い、得られた結果から情報を得ます。この方法のポイントは、実際の計算をゲートがになっているということです。

一方これに対して、測定型量子計算 (Measurement Based Quantum Computation)(*MBQC*) は、全く異なる発想のもと計算を行います。表1にあるように、まず十分にエンタングルした量子状態をはじめに準備しておき、これに何度も測定していき量子ビットを破壊していくことで変換が進みます。つまり、計算を担っているのが測定行為なのです。この新しい計算方法の発見により、量子コンピュータに関して理論・実験の両面で様々な進展があり、現在も研究が行われています。今回はその測定型量子計算にスポットを当てて、紹介していきます。

前提知識としては、量子状態のユニタリー変換に関する簡単な計算や量子力学における射影測定の計算を多少知っている人を対象としています。量子力学に少し触れたことがある程度の人でも概要がわかるように努めました。

ゲート型	測定型

表 1: (左) ゲート型量子計算。エンタングルしていない初期状態を準備し、ゲートを次々にかけていく。測定は破壊的操作のため、基本的には最後に各量子ビットに対して 1 回づつしか行わない。(右) 測定型量子計算。白丸は量子ビットを表す。エンタングルした初期状態を初めから準備し、測定を繰り返すことでやりたい量子計算を行う。測定によって状態が変わってしまうことをうまく利用している。

2 約束および基本事項のまとめ

このセクションでは、量子計算における重要事項のまとめと、本稿で使用するノーテーションについて整理しておきます。

量子計算において、観測の対象となるものは量子ビットと呼ばれるものです。この量子ビットの状態はベクトルで表現されます。また状態の変換はベクトルの変換として解釈され、行列を用いて記述することができます。この変換は、(1) 線型性、(2) ユニタリー性の二つの条件を満たさなければいけません。

それらの条件を満たす変換行列のうち、量子計算にとって最も重要なパウリ行列は、次のように表されます。

$$X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \quad Y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix} \quad Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} \quad (1)$$

これらの行列のうち Z の固有値 $1, -1$ の固有状態をそれぞれ $|0\rangle, |1\rangle$ と表記します。つまり、

$$|0\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \quad (2)$$

$$|1\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix} \quad (3)$$

となります。これらの状態を計算基底と呼びます。

さらに重要な基底として、 X の固有値 $1, -1$ の固有状態をそれぞれ $|+\rangle, |-\rangle$ と表記します。計算基底を用いて書くと、

$$|+\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) \quad (4)$$

$$|-\rangle = \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle) \quad (5)$$

と表されます。

本稿において、測定は PVM 測定 (射影測定) のことを指します。PVM 測定は基底の組 $\{|i\rangle\}$ に対して定義され、状態 $|\phi\rangle$ に対する測定後の状態が

$$|\phi\rangle \langle i|\phi\rangle \quad (6)$$

になるものを指します*1。ただし、どの基底 $|i\rangle$ が選ばれるかは、確率的に決まります。これを人間がコントロールすることはできません。また、量子力学において測定結果は演算子の固有値と対応しますが、量子計算の文脈においてよく用いられるように、本稿においては測定値が 1 になったとき「結果が 0 になった」と呼び、測定値が -1 になったとき「結果が 1 になった」と呼ぶことにします。

最後に、いくつかの重要な変換 (ゲート) の作用を行列表示を使って記述すると、次のようになります。

$$R_x(\theta) = \begin{pmatrix} \cos \frac{\theta}{2} & i \sin \frac{\theta}{2} \\ -i \sin \frac{\theta}{2} & \cos \frac{\theta}{2} \end{pmatrix} \quad R_z(\theta) = \begin{pmatrix} \exp(-i\frac{\theta}{2}) & 0 \\ 0 & \exp(i\frac{\theta}{2}) \end{pmatrix} \quad (7)$$

$$CZ = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & -1 \end{pmatrix} \quad (8)$$

$$CX = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} \quad (9)$$

CZ, CX ゲートは 2 量子ビットの状態に対する作用であり、行列のテンソル積を利用して記述しています。

*1 厳密には、測定とは状態から確率分布を定める演算子のこと指します。しかし今回は特定の状態が選ばれる確率よりも測定後の状態の方が興味があるため、どのような確率分布になるかについては言及しません。

3 測定型量子計算の基礎

ここから、測定型量子計算の概念を導入します。ここでの導入は [4] に従います。

3.1 コマンドとパターン

測定型量子計算の操作は、コマンド列によって構成されるパターンによって記述されます。まずはこのコマンドとパターンを定義します。

Definition 3.1 (コマンド). $s, t \in \{0, 1\}$ とする。コマンドとは、次に当たられる操作のことである。

1. N_i : i 番目の量子ビットを $|+\rangle$ として準備する。
2. E_{ij} : i 番目と j 番目の量子ビットを CZ でエンタングルさせる。
3. ${}^t[M_i^\alpha]^s$: i 番目の量子ビットを角度 $(-1)^s\alpha + t\pi$ で測定する。
4. X_i^s : i 番目の量子ビットにパウリ演算子 X^s を作用させる。
5. Z_i^s : i 番目の量子ビットにパウリ演算子 Z^s を作用させる。

s や t が 0 とわかっているとき、それを省略することがある。(例えば、 ${}^0[M_i^\alpha]^0 = [M_i^\alpha]^0 = M_i^\alpha$)。また、 s, t には基本的に添字をつけて s_i, t_i などと表記する。その添字をつけた s_i, t_i は、 i 番目の量子ビットの測定結果を表す。

ここで、いくつか注意すべきことがあります。まず、「 CZ でエンタングルさせる」とは、 i 番目と j 番目の量子ビットにコントロール Z ゲートを作用させることを意味します。ここで、 CZ ゲートはどちらを制御ビットにとっても結果は変わらないため、 $CZ |i\rangle \otimes |j\rangle$ という表現が許されます。

また、「角度 α で測定する」とは、基底

$$|+\alpha\rangle = \frac{1}{\sqrt{2}}(|0\rangle + e^{i\alpha}|1\rangle) \quad (10)$$

$$|-\alpha\rangle = \frac{1}{\sqrt{2}}(|0\rangle - e^{i\alpha}|1\rangle) \quad (11)$$

を使って射影測定をするという意味です。当然測定結果は確率的に決まるので、コマンドによって定まる量子計算の結果は一意ではありません。このことが測定型量子計算の 1 つのポイントでもあります。

この性質を用いて、すでに測定した量子ビットの測定結果を次の操作に利用することもあり得ます。例えば i 番目の量子ビットの測定で $|+\rangle$ が選ばれたとき、 $s_i = 1$ であり、 $X_j^{s_i} = X_j$ は「 j 番目の量子ビットに X を作用させる」ことを意味します。また逆に i 番目の量子ビットの測定で $|-\rangle$ が選ばれたとき、 $s_i = 0$ であり、 $(X_j)^2$ は恒等演算子と等しくなり、「 j 番目の量子ビットに何もしない」ことを意味します。後ほどさらに詳しい具体例を考えます。

このコマンドと呼ばれる操作を複数回繰り返すコマンド列のことを、パターンと呼びます。詳細な定義は、次の通りです。

Definition 3.2 (パターン). パターンとは、次の条件を満たす有限コマンドの列、および有限集合の組 (V, I, O) のことである。

1. まだ測定されていない結果をコマンドに利用してはならない。
2. すでに測定された量子ビットに操作をしてはならない。
3. まだ準備されていない量子ビットに操作をしてはならない。
4. 集合 O に属さない量子ビットは測定されなければならない。

ここに、 V は量子ビットの集合、 I は入力に対応する量子ビットの集合、 O は出力に対応する量子ビットの集合を表す。また、特にこのコマンド列から成る操作のことを、測定型量子計算と呼ぶ。

ここで、コマンドとパターンの意味をみるために、具体例を確認してみます。以下では、コマンド列は右から順番に働くことを注意してください。

Example 3.3 (パターンではない例). $V = \{1, 2, 3\}, I = \{1\}, O = \{2, 3\}$ とする。コマンド列

$$M_1^\pi X_2^{s_1} E_{32} E_{12} N_3 N_2 \quad (12)$$

は、1 番目の量子ビットに対する測定 M_1^π を行う前に、その結果を用いる演算 $X_2^{s_1}$ を行ってしまっているため、パターンの条件 1 を満たさず、パターンではない。

Example 3.4 (パターンの例). $V = \{1, 2\}, I = O = \{1\}$ とする。このとき、コマンド列

$$\Sigma = X_1^{s_2} M_2^0 E_{12} N_2 \quad (13)$$

は、パターンの満たすべき条件 1~4 を全て満たすため、パターンである。

上記の例から分かるように、初めにこれから使う全ての量子ビットで N コマンドを行うことは自明であるため、以降は量子ビットの準備を表す N コマンドは省略します。

さて、コマンド列からなるパターンを定義しました。これが量子計算として実際に何を行なっているかについて確認してみましょう。その前に、パターンが量子状態を量子状態に移す関数であることを確認します。

Remark 3.5. 定義 3.2 より、集合 I に属する量子ビットの量子状態を初期状態とみなし、集合 O に属する量子ビットの量子状態を終状態とみなすことができる。これにより、コマンドは量子演算とみなすことができる。この関数はユニタリ演算であることが証明できる。

では、例 3.4 がどんなユニタリ演算であるかを実際に確認してみます。入力に対応する量子

ビットの初期状態を $a|0\rangle + b|1\rangle$ とおくと、

$$X_1^{s_2} M_2^0 E_{12} N_2 (a|0\rangle_1 + b|1\rangle_1) \quad (14)$$

$$= X_1^{s_2} M_2^0 E_{12} (a|0\rangle_1 + b|1\rangle_1) \frac{1}{\sqrt{2}} (|0\rangle_2 + |1\rangle_2) \quad (15)$$

$$= X_1^{s_2} M_2^0 \frac{1}{\sqrt{2}} (a|00\rangle_{12} + a|01\rangle_{12} + b|10\rangle_{12} - b|11\rangle_{12}) \quad (16)$$

$$= X_1^{s_2} M_2^0 \frac{1}{\sqrt{2}} (a(|++\rangle_{12} + |+-\rangle_{12}) + b(|+-\rangle_{12} - |--\rangle_{12})) \quad (17)$$

$$= \begin{cases} X_1^0 a |0\rangle_1 \\ X_1^1 b |1\rangle_1 \end{cases} \quad (18)$$

$$= \begin{cases} a |0\rangle_1 \\ b |0\rangle_1 \end{cases} \quad (19)$$

と計算されます。

この計算の流れは、次の通り説明されます。まず、量子ビット 1 を用意します。そして $|+\rangle$ の量子ビット 2 を用意します。そして二つの量子ビットを CZ ゲートを用いてエンタングルさせます。その後、2 番目の量子ビットを決められた基底で測定します。そして実験者はその測定結果を見て、1 番目の量子ビットに X 演算をするかどうかを決めます。

この計算結果から、全ての入力に対して (規格化すると) 必ず状態 $|0\rangle$ が返されることがわかりました。この例のように位相因子を除いて全ての場合において同じ結果が出力される時、そのパターンを決定的と呼びます。また、さらに条件を強くして、全ての場合において位相因子も含めて同じ結果が出力される時、強決定的と呼びます。どのパターンが決定的、または強決定的かを判断するのは重要な問題であり*2、決定的でないパターンももちろん存在します。

3.2 ユニバーサリティー

さて、一部のコマンドは初期状態から終状態に状態を一意に変換する関数の役割を果たしていることを確認しました。逆に、我々がしたい変換を表す「決定的な」パターン*3は常に存在するのでしょうか。もし存在すれば、運に左右されず任意の量子計算を実行できることになるので、パターンによる測定型量子計算は通常のゲート型量子計算と全く等価であると言えます。この問いには肯定的な答えが存在します。

Theorem 3.6 (ユニバーサリティセット). $\otimes^n \mathbb{C}^2$ 上の任意のユニタリーゲートは、 $J(\alpha)$ と CZ の組み合わせで実現できる。ただし

*2 この問題はすでに解決されており、パターンをグラフ化したものに対して、決定的なパターンが存在するための必要十分条件は「g-Flow」と呼ばれるグラフに対する条件を用いて判断されることが証明されています。

*3 もし決定的でなければ、計算を行うたびに異なる状態が出力されることを意味します。それは不便なので、ここでは「何回計算をしても状態を同じように変換する」ことを目指します。

$$J(\alpha) = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & e^{i\alpha} \\ 1 & -e^{i\alpha} \end{pmatrix} \quad (20)$$

である。

Theorem 3.7 (測定型量子計算のユニバーサリティー). 定義 3.2 で定義される測定型量子計算は、ユニバーサルである。

Proof. $V = \{1, 2\}, I = \{1\}, O = \{2\}$ とする。ユニタリーゲート $J(\alpha), CZ$ は、

$$J(\alpha) = X_2^{s_1} M_1^{-\alpha} E_{12} \quad (21)$$

$$CZ = E_{12} \quad (22)$$

というコマンド列で実現される。(実際、入力状態を $a|0\rangle + b|1\rangle$ とおくと、

$$X_2^{s_1} M_1^{-\alpha} E_{12} (a|0\rangle_1 + b|1\rangle_1) \frac{1}{\sqrt{2}} (|0\rangle_2 + |1\rangle_2) \quad (23)$$

$$= X_2^{s_2} M_1^{-\alpha} \frac{1}{\sqrt{2}} (a|00\rangle_{12} + a|01\rangle_{12} + b|10\rangle_{12} - b|11\rangle_{12}) \quad (24)$$

$$= \begin{cases} X_2^0(\frac{1}{2})[(a + be^{i\alpha})|0\rangle_2 + (a - be^{i\alpha})|1\rangle_2] \\ X_2^1(\frac{1}{2})[(a - be^{i\alpha})|0\rangle_2 + (a + be^{i\alpha})|1\rangle_2] \end{cases} \quad (25)$$

$$= \begin{cases} \frac{1}{2}[(a + be^{i\alpha})|0\rangle_2 + (a - be^{i\alpha})|1\rangle_2] \\ \frac{1}{2}[(a + be^{i\alpha})|0\rangle_2 + (a - be^{i\alpha})|1\rangle_2] \end{cases} \quad (26)$$

$$(27)$$

となり、強決定的に状態が決まる。これを行列を用いて書くと、確かに

$$X_2^{s_1} M_1^{-\alpha} E_{12} = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & e^{i\alpha} \\ 1 & -e^{i\alpha} \end{pmatrix} \quad (28)$$

となる。) この結果と定理 3.6 より、測定型量子計算はユニバーサルである。□

以上の結果より、測定型量子計算は任意のユニタリー量子計算を実行できることが確認されました。そもそも我々は量子ビットに対する操作に対してユニタリー性を課していたため、測定型量子計算はあらゆる操作を決定的に再現できることがわかります。つまり、測定型量子計算は通常のゲート型量子コンピュータと同じ計算能力を持っているということです。

3.3 パターンの並びかえ

しかし一般に、コマンド E, M, X, Z がバラバラに並んでいるパターンは扱いにくいですし、何をやっているのかが見えてきません。次は、このコマンドを特定の順番に並び替えることができることを証明します。

Proposition 3.8 (EMC 形への変形). 任意のパターンは、 E の後に M を操作し、 M の後に C を操作する、関数として同一なパターンに変形できる。ただし C とは X または Z のことである。

Remark 3.9. EMC 形とは $E \rightarrow M \rightarrow C$ の順番で操作を行うことを指すので、本稿が採用している「右から順番に量子ビットに作用する」というコマンド列の書き方では、 CME の順番になることに注意する。

Proof. 次の書き換え規則が成立する。

$$E_{ij}X_i^s \rightarrow X_i^sZ_j^sE_{ij} \quad (29)$$

$$E_{ij}X_j^s \rightarrow X_j^sZ_i^sE_{ij} \quad (30)$$

$$E_{ij}Z_i^s \rightarrow Z_i^sE_{ij} \quad (31)$$

$$E_{ij}Z_j^s \rightarrow Z_j^sE_{ij} \quad (32)$$

$${}^t[M_i^\alpha]^s X_i^r \rightarrow {}^t[M_i^\alpha]^{s+r} \quad (33)$$

$${}^t[M_i^\alpha]^s Z_i^r \rightarrow {}^{r+t}[M_i^\alpha]^s \quad (34)$$

これらの関係式が成立することは、行列計算より確認することができる。また、異なる量子ビットに作用するコマンドは可換である。

さて、コマンド列が EMC 形になっていなかったとする。その部分列の中には、 EM 、 EC 、 MC のうちいくつかが存在する。これらを次のように書き換えることで、 EMC 形に変形することができる。

1. EM に対しては、パターンの定義より、 E と M の添え字になる量子ビットは異ならなければならない。したがって、この場合は「異なる量子ビットに作用するコマンドは可換である」ことより ME に書き換えることができる。
2. EC に対しては、 E と C の添え字が異なるとき、可換である。添え字に共通のものがあるとき、上記の書き換え規則を用いて CE または CCE に変形できる。
3. MC に対しては、 M と C の添え字が異なるとき、可換である。添え字に共通のものがあるとき、書き換え規則により M に書き換えることができる。

□

この事実より、コマンドを EMC 形に変形することで、あらゆるコマンド操作を次のように統一的な描像で理解することができます。まずコマンド E に対応して、状態 $|+\rangle$ の量子ビットのいくつかをエンタングルさせます (グラフ状態の準備)。次にコマンド M に対応して、各量子ビットに測定をしていきます (測定)。このとき測定角度は、今までの測定結果に応じて決めていきます。最後に、測定で求めていた結果が得られなかった回数に応じて、コマンド C に対応する X または Z ゲートをかけます (副次的演算子の作用)。この一連の行為が測定型量子計算で行うべきことの全てです。

3.4 いくつかの例とグラフによる描像

測定型量子計算がゲート型と同じ計算能力を持つことが確認されました。それでは、量子回路でよく出てくる基本的な演算は、具体的にどのようなコマンド列に変換されるのでしょうか。また、

これらのコマンド列をグラフで描像することも考えてみます。いくつかの例をみてみましょう。

x 軸周りの回転 $R_x(\theta)$

x 軸周りの回転 $R_x(\theta)$ を表すパターンは、 $V = \{1, 2, 3\}, I = \{1\}, O = \{3\}$ として、

$$X_3^{s_2} Z_3^{s_1} [M_2^{-\theta}]^{s_1} M_1^0 E_{23} E_{12} \quad (35)$$

で表されます。実際にこれが x 軸周りの回転を表現していることは、単純な計算により次のように確かめられます：

初期状態 $a|0\rangle + b|1\rangle$ に対して、

$$\begin{aligned} & X_3^{s_2} Z_3^{s_1} [M_2^{-\theta}]^{s_1} M_1^0 E_{23} E_{12} (a|0\rangle_1 + b|1\rangle_1) \frac{1}{\sqrt{2}}(|0\rangle_2 + |1\rangle_2) \frac{1}{\sqrt{2}}(|0\rangle_3 + |1\rangle_3) \\ = & X_3^{s_2} Z_3^{s_1} [M_2^{-\theta}]^{s_1} M_1^0 \frac{1}{2} (a|000\rangle_{123} + a|001\rangle_{123} \\ & + a|010\rangle_{123} - a|011\rangle_{123} + b|100\rangle_{123} + b|101\rangle_{123} - b|110\rangle_{123} + b|111\rangle_{123}) \\ = & \begin{cases} X_3^{s_2} Z_3^{s_1} [M_2^{-\theta}]^{s_1} \frac{1}{2\sqrt{2}} \{ (a+b)|00\rangle_{23} + (a+b)|01\rangle_{23} \\ + (a-b)|10\rangle_{23} + (-a+b)|11\rangle_{23} \} & (s_1 = 0) \\ X_3^{s_2} Z_3^{s_1} [M_2^{-\theta}]^{s_1} \frac{1}{2\sqrt{2}} \{ (a-b)|00\rangle_{23} + (a-b)|01\rangle_{23} \\ + (a+b)|10\rangle_{23} + (-a-b)|11\rangle_{23} \} & (s_1 = 1) \end{cases} \\ = & \begin{cases} X_3^{s_2} Z_3^{s_1} \frac{1}{4} \{ ((a+b) + (a-b)e^{i\theta})|0\rangle_3 + ((a+b) + (-a+b)e^{i\theta})|1\rangle_3 \} ((s_1, s_2) = (0, 0)) \\ X_3^{s_2} Z_3^{s_1} \frac{1}{4} \{ ((a+b) - (a-b)e^{i\theta})|0\rangle_3 + ((a+b) - (-a+b)e^{i\theta})|1\rangle_3 \} ((s_1, s_2) = (0, 1)) \\ X_3^{s_2} Z_3^{s_1} \frac{1}{4} \{ ((a-b) + (a+b)e^{-i\theta})|0\rangle_3 + ((a-b) - (a+b)e^{-i\theta})|1\rangle_3 \} ((s_1, s_2) = (1, 0)) \\ X_3^{s_2} Z_3^{s_1} \frac{1}{4} \{ ((a-b) - (a+b)e^{-i\theta})|0\rangle_3 + ((a-b) + (a+b)e^{-i\theta})|1\rangle_3 \} ((s_1, s_2) = (1, 0)) \end{cases} \\ = & \begin{cases} X_3^{s_2} \frac{1}{4} \{ ((a+b) + (a-b)e^{i\theta})|0\rangle_3 + ((a+b) + (-a+b)e^{i\theta})|1\rangle_3 \} ((s_1, s_2) = (0, 0)) \\ X_3^{s_2} \frac{1}{4} \{ ((a+b) - (a-b)e^{i\theta})|0\rangle_3 + ((a+b) - (-a+b)e^{i\theta})|1\rangle_3 \} ((s_1, s_2) = (0, 1)) \\ X_3^{s_2} \frac{1}{4} \{ ((a-b) + (a+b)e^{-i\theta})|0\rangle_3 + ((-a+b) + (a+b)e^{-i\theta})|1\rangle_3 \} ((s_1, s_2) = (1, 0)) \\ X_3^{s_2} \frac{1}{4} \{ ((a-b) - (a+b)e^{-i\theta})|0\rangle_3 + ((-a+b) - (a+b)e^{-i\theta})|1\rangle_3 \} ((s_1, s_2) = (1, 1)) \end{cases} \\ = & \begin{cases} \frac{1}{4} \{ ((a+b) + (a-b)e^{i\theta})|0\rangle_3 + ((a+b) + (-a+b)e^{i\theta})|1\rangle_3 \} ((s_1, s_2) = (0, 0)) \\ \frac{1}{4} \{ ((a+b) + (a-b)e^{i\theta})|0\rangle_3 + ((a+b) + (-a+b)e^{i\theta})|1\rangle_3 \} ((s_1, s_2) = (0, 1)) \\ \frac{1}{4} \{ ((a-b) + (a+b)e^{-i\theta})|0\rangle_3 + ((-a+b) + (a+b)e^{-i\theta})|1\rangle_3 \} ((s_1, s_2) = (1, 0)) \\ \frac{1}{4} \{ ((-a+b) - (a+b)e^{-i\theta})|0\rangle_3 + ((a-b) - (a+b)e^{-i\theta})|1\rangle_3 \} ((s_1, s_2) = (1, 1)) \end{cases} \\ = & \begin{cases} \frac{1}{4} \{ ((a+b) + (a-b)e^{i\theta})|0\rangle_3 + ((a+b) + (-a+b)e^{i\theta})|1\rangle_3 \} ((s_1, s_2) = (0, 0)) \\ \frac{1}{4} \{ ((a+b) + (a-b)e^{i\theta})|0\rangle_3 + ((a+b) + (-a+b)e^{i\theta})|1\rangle_3 \} ((s_1, s_2) = (0, 1)) \\ \frac{e^{-i\theta}}{4} \{ ((a+b) + (a-b)e^{i\theta})|0\rangle_3 + ((a+b) + (-a+b)e^{i\theta})|1\rangle_3 \} ((s_1, s_2) = (1, 0)) \\ -\frac{e^{-i\theta}}{4} \{ ((a+b) + (a-b)e^{i\theta})|0\rangle_3 + ((a+b) + (-a+b)e^{i\theta})|1\rangle_3 \} ((s_1, s_2) = (1, 1)) \end{cases} \end{aligned}$$

と最終状態が位相因子を除いて決定的に決まる。

このコマンド列をグラフで表現するために、次のような規則を取り入れます*4。

- 各頂点は V の元に対応する。
- コマンド列に E_{ij} が存在するとき、 i, j を線で結ぶ。

*4 色分けについての統一的な決まりはなく、本稿独自の定義です。

- 測定する量子ビットに対応する頂点のうち、 x 基底での測定はオレンジ色で、 y 基底での測定は緑色で表現する。それ以外の測定は赤色で表現する。
- アウトプットに対応する頂点には色をつけない。

この規則に基づいて式 35 をグラフ化すると表 2a のようになります。

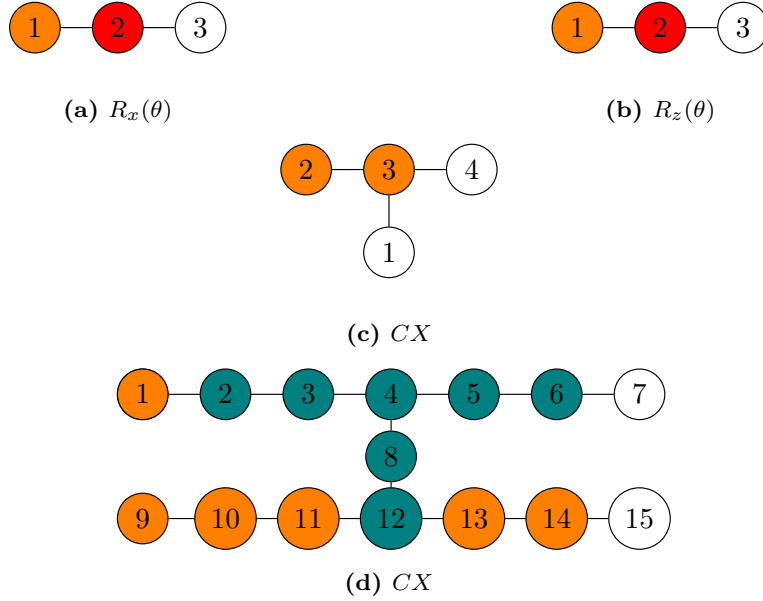


表 2: パターンのグラフ表現。それぞれ (a) x 軸周りの角度 θ の回転。(b) z 軸周りの角度 θ の回転。(c)(d) CX ゲート。グラフとして表現すると副次演算子の作用が現れないが、コマンド列を EMC 形に変形しているので、全ての測定が終わった後に (測定が失敗した分だけ*5) 副次演算子を作用させれば良い。

z 軸周りの回転 $R_z(\theta)$

同様に z 軸周りの回転を表す $R_z(\theta)$ は、具体的な確認は省略しますが、 $V = \{1, 2, 3\}, I = \{1\}, O = \{3\}$ として、

$$X_3^{s_2} Z_3^{s_1} M_2^0 M_1^{-\alpha} E_{23} E_{12} \quad (36)$$

で与えられます。これを同様の規則でグラフにすると表 2b のようになります。

CZ ゲート

CZ ゲートのパターン実装は簡単です。 $V = \{1, 2\}, I = \{1, 2\}, O = \{1, 2\}$ として、

$$E_{12} \quad (37)$$

*5 ここでいう「失敗」とは、実験に失敗してうまく測定できなかったことを意味するのではなく、測定結果が「1」になってしまうことを指します。

とすれば良いだけです。

CX ゲート

CX ゲートの実装はやや難しいですが、結果だけ与えると、 $V = \{1, 2, 3, 4\}$, $I = \{1, 2\}$, $O = \{1, 4\}$ として、

$$X_4^{s_3} Z_4^{s_2} Z_1^{s_2} M_3^0 M_2^0 E_{13} E_{23} E_{34} \quad (38)$$

というパターンでうまくいくことが確認されます。グラフは表 2c のようになります。

グラフの形状や実験方法などに応じて、異なるパターンを考える方が良い場合もあります。例えば、グラフの形を変えて表 2d のようなグラフも、CX ゲートを再現します。

4 MB-VQE

ここまでで、測定型量子計算とはどのような計算であるのかを、数式を用いて解説しました。ここからはこの計算手法を使って実際に何か意味のある計算を試みることを考えます。その一例として、測定型の考え方から発想を得た量子計算である MB-VQE[5] について紹介します。

変分量子固有値ソルバー (Variational Quantum Eigensolver : VQE) とは、日本語では変分量子固有値ソルバーと呼び、ハミルトニアン基底状態の固有値を量子回路を用いて計算するアルゴリズムのことを指します。その量子計算の測定型バージョンが測定型変分量子固有値ソルバー (Measurement Based Variational Quantum Eigensolver : MB-VQE) です。この手法を、実際にいくつかの計算例に触れながら紹介します。

4.1 MB-VQE の手法 (1)

まず、通常のゲート型でハミルトニアンの固有値を計算するにはどのようにすればいいかを確認しましょう。

ハミルトニアン H の基底状態を求めるために、適当な初期状態を入力とし、各ビットに y 軸周りの回転ゲートと z 軸周りの回転ゲート*6を作用させます (表 3a)。このとき、回転角度も最初は適当に設定します。その後、各隣接するビットに CZ ゲートを作用させます。得られた結果の量子状態 $|\psi\rangle$ からハミルトニアンの期待値

$$\langle \psi | H | \psi \rangle \quad (39)$$

を計算します。任意の規格化された状態 $|\psi\rangle$ に対して式 (39) は基底状態で最小値 E_0 を取るこ

*6 量子回路の作り方は経験的なものであり、必ずしもこの通りである必要はありません。例えば、表 3a では 4 ビットの VQE のための回路の例を示しており、パラメータを 8 つ有していますが、点線で囲まれた部分を 1 セットとして繰り返すことで、パラメータの数をさらに増やすことも考えられます。他にも、CZ ゲートを任意の 2 量子ビットの組み合わせに対して作用させたり、回転の軸や個数を変えることも工夫として存在します。

が知られています。つまり、

$$\min_{|\psi\rangle} \langle \psi | H | \psi \rangle = E_0 \quad (40)$$

となります。

Theorem 4.1 (変分法). E_0 をハミルトニアン H の基底状態とする。任意の $\langle \psi | \psi \rangle = 1$ を満たす状態 $|\psi\rangle$ について、

$$\langle \psi | H | \psi \rangle \geq E_0 \quad (41)$$

が成立する。等号成立は $|\psi\rangle$ が基底状態のときである。

Proof. $|n\rangle$ をハミルトニアンの固有状態とする。 $|\psi\rangle$ を固有状態を用いて展開すると、

$$|\psi\rangle = \sum_{n=0} c_n |n\rangle \quad (42)$$

となる。ただし $\langle \psi | \psi \rangle = 1$ より $\sum_{n=0} |c_n|^2 = 1$ である。このとき、

$$\langle \psi | H | \psi \rangle = \sum_{m=0} \sum_{n=0} c_m^* c_n \langle m | H | n \rangle = \sum_{n=0} |c_n|^2 E_n \geq E_0 \sum_{n=0} |c_n|^2 = E_0 \quad (43)$$

□

この定理を用いて、VQE の流れは次のように表現できます:

- 1 適当に初期状態と回路パラメータを定め計算をする。
- 2 得られた結果に対してハミルトニアンの期待値測定を行う。
- 3 測定の結果がより小さくなるように、パラメータの値を調整する。
- 4 再度量子計算を行う。
- 5 以後期待値測定の結果が最小になるまで繰り返す。

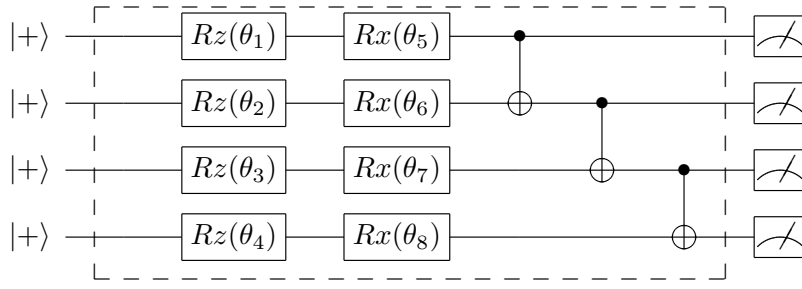
このように計算を進めると、基底状態のエネルギー固有値を得ることができます。

ここまですがゲート型の VQE の流れです*7。それでは、MB-VQE への応用として、測定型量子計算がゲート型と書き換え可能であることを利用して、VQE の回路をそのまま MBQC に書き換えることをここでは考えてみます。

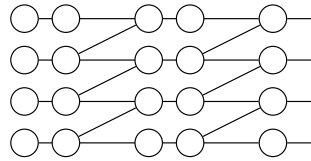
表 3a で描かれた回路は先ほど説明した VQE のための量子回路であり、 R_z, R_x, CX ゲートから構成されます。これらのコマンド列への変換は、それぞれ表 1 のように与えられました。量子状態に次々とゲートを作用させることは、グラフ上においてアウトプットの頂点を次のインプットの頂点としてグラフをつなげていくことに対応します。

ここでは単にグラフを繋げるだけではなく、別の工夫をしてみます。そのための準備として、次の定理が有用です。

*7 より詳細は [1] にあります。



(a) VQE の回路図



(b) 対応する MB-VQE

表 3: (a)VQE の回路図。 R_x, R_z は y, z 軸周りの角度 θ の回転。ループ数に応じて点線で囲まれた部分が繰り返される。(b) 対応する MB-VQE。図は回路のループ数が 2 のとき。

Theorem 4.2 (Gottesman-Knill の定理). 計算基底の状態の準備、クリフォードゲート^{*8}の作用、および計算基底での測定は、古典計算機で効率的にシミュレートできる。

この定理の存在により、クリフォードゲートなど一部の量子計算は、そもそも量子コンピュータを持ち出す必要がなく、古典コンピュータで十分であることがわかります。これを利用して、クリフォード群に属する計算をグラフから取り除いて (表 2 のオレンジ色と緑色の部分)、グラフを連結させたのが、表 3b です。これをみてもわかるように、ループ回数が 2 のときゲート型では 16 個の回転操作と 3 つの 2 量子ビットゲート操作を必要とするのに対し、測定型では 20 個の量子ビットと 16 回の測定が必要になります。このように記述すると効果はわかりにくいですが、元の VQE 回路にクリフォードゲートが多かった場合などは特に、量子コンピュータで行う必要がある計算量を減らすことができます。この方法はゲート型の量子計算を測定型に翻訳しただけなので、理想的には二つのやり方は等価であることに注意してください。

^{*8} クリフォード群とは、H ゲート、CX ゲート、S ゲートを生成子とする群のことである。ここではパウリゲート (X,Y,Z ゲートのこと) と CX,CZ ゲート、パウリ基底での測定がクリフォード群に入り、一般の角度の回転を表すゲートはクリフォード群に入らないことさえ知っていれば良い。

例:量子イジング模型

さて、実際に MB-VQE を用いて量子計算を行なった例を確認します。まず、ハミルトニアンとして、量子イジング模型

$$H = \sum_{i=1}^{n-1} Z_i Z_{i+1} + \lambda \sum_i^n X_i \quad (44)$$

を取ったときの基底状態を求める問題を解いてみましょう。ここにハミルトニアンの第二項は一様磁場の効果を表します。ループ回数と摂動の大きさ λ を変えて、基底状態のエネルギー期待値を計算した結果を厳密解と比較したものが図 1 になります。実際に、ループ回数を 2 にすると、どちらの場合でも厳密解とよく一致^{*9}していることが確認されます。

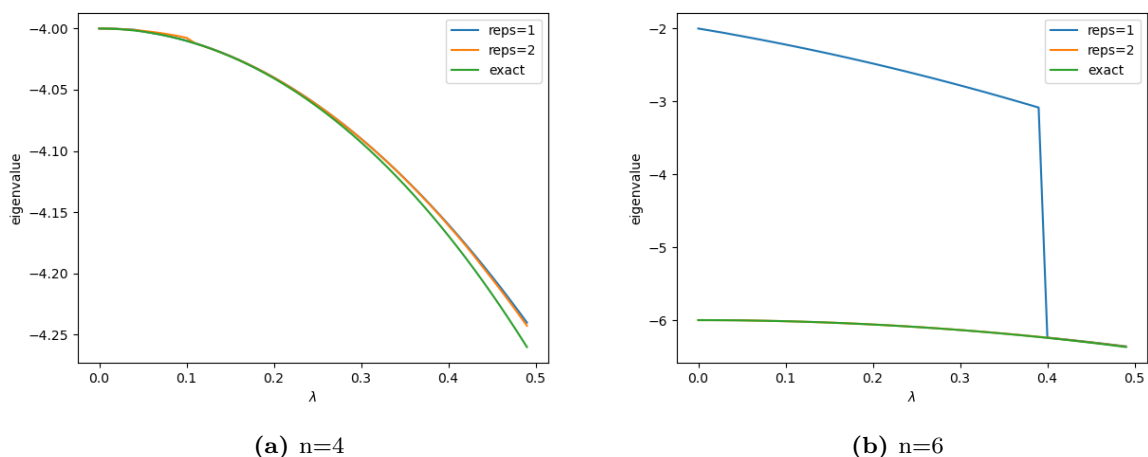


図 1: 量子イジングモデルの固有値。reps は VQE の回路の繰り返し回数、exact は対角化による厳密解。(a) $n=4$ 、(b) $n=6$ 。

例:頂点被覆問題

次の例として、NP 困難であることが知られる頂点被覆問題を MB-VQE で調べてみましょう。(参考: [2][3])

頂点被覆問題とは、次のように定式化されます。

頂点被覆問題

グラフ $G = (V, E)$ に対して、条件「任意の辺について、両端の頂点のうち少なくとも一つが A に属する」を満たす頂点集合 $A \in V$ のうち要素数が最小のものを求める。

^{*9} $n=6$ かつ繰り返し回数が 1 の時、 λ の値が小さい範囲では両者はあまり一致していません。これは基底状態を導くまでに必要な回路の数が少なすぎることが関係していると考えられます。

図2のグラフを例にとると、解となる集合 A は $A = \{1, 3, 5\}$ または $\{1, 2, 5\}$ であり、その要素数は3です。

これを MB-VQE で解くために、次のステップを行います。

- 1 頂点被覆問題を古典イジング最適化問題に還元する。
- 2 古典イジング模型を量子イジング模型に還元する。
- 3 MB-VQE を用いて最適化を行い、基底状態を求める。
- 4 得られた基底状態から古典イジング模型の基底状態を推定する。

それぞれのステップについて解説します。まずステップ1では、次のハミルトニアンを定義します。

$$H_c = L \sum_{uv \in E} (1 - x_u)(1 - x_v) + M \sum_{v \in V} x_v \quad (45)$$

(ただし x_i は 0,1 を値としてとりうるバイナリー変数であり、 L, M は $L \gg M$ を満たす定数。) このハミルトニアンは、集合 A に属する要素のみが1、それ以外が0となる値の系列 $x_1^* x_2^* \cdots x_n^*$ に対して最小化されることが容易に確認されます*10。

次にステップ2では、これを量子イジングハミルトニアンに書き換えます:

$$H_q = L \sum_{u,v \in E} \left(1 - \frac{1 - Z_u}{2}\right) \left(1 - \frac{1 - Z_v}{2}\right) + M \sum_{v \in V} \frac{1 - Z_v}{2} \quad (46)$$

これは古典ハミルトニアンの x_u をパウリ演算子 $\frac{1 - Z_u}{2}$ に書き換えただけですが、この書き換えによって、ハミルトニアンの基底状態が $|x_1^* x_2^* \cdots x_n^*\rangle$ という量子状態になってくれます。

次にステップ3では、これを MB-VQE で実装します。今回は [3] で回路作成の関数が与えられており、それに基づいてシミュレーションをしました(図3)。3bと同様に x 回転、 z 回転、CZ を繰り返すモデルとなっています。

ハミルトニアンがどのように回路(グラフ)として実装されるのかは自明ではありませんので、補足しておきます。まず、変数の数だけ(今回の場合は頂点の数)縦に量子ビットを並べます。この5つ組を1セットとして相互に変換しながら(辺でつなげながら)横に並べていくことで、目的の量子状態を作っていきます。ここで、「相互の変換」のために、被覆問題のグラフの2つの頂点に辺がつながっている部分にのみ、図3の白い量子ビットで与えられた量子変換を行います。このようにして、グラフは量子回路という別のグラフに変換されます。

最後にステップ4では、得られた量子状態から古典的なビット系列で表される解を推定します。VQEによって得られた基底状態は状態は近似解ですのでいくつかの状態の重ね合わせになっているはずですが、そのため得られた状態に対して射影測定を行い、最もらしい基底を決める必要があります。

*10 第1項は、変数の系列が「任意の辺について、両端の頂点のうち少なくとも一つが A に属する」という条件を満たす時のみ0になり、これを満たさない場合エネルギーは正の値を取ります。第2項は、集合 A の要素が少ないほど小さな値を取ります。この事実から、エネルギーを最小化する系列が頂点被覆問題の解となります。ただし、条件を満たすことが A の要素数が少ないことより重要なため、定数 L は M より十分大きい必要があります。

図 4 は、以上の操作を経て 2 で示されたグラフについて MB-VQE を実行した結果です。確かに解となる状態 $|10101\rangle, |11010\rangle$ で状態の確率振幅が高くなっていることがわかります。この結果から、頂点被覆問題の解が $A = \{1, 3, 5\}$ や $\{1, 2, 5\}$ であることが求められました。

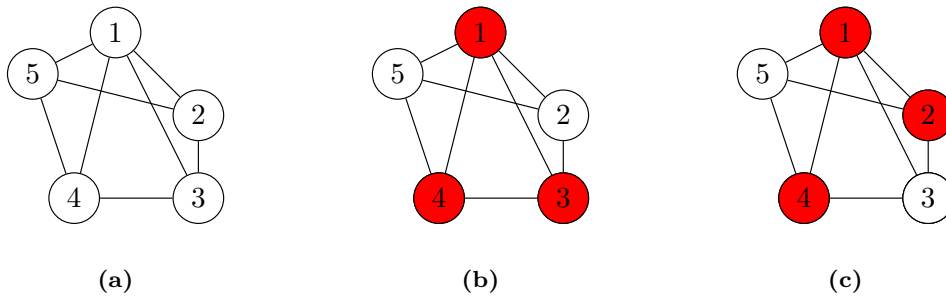


図 2: 頂点被覆問題を考えるグラフ。(a) のグラフの頂点被覆をする最小の部分集合は (b) の赤丸の集合である。解は一つとは限らず、(c) のようなものも解になる。

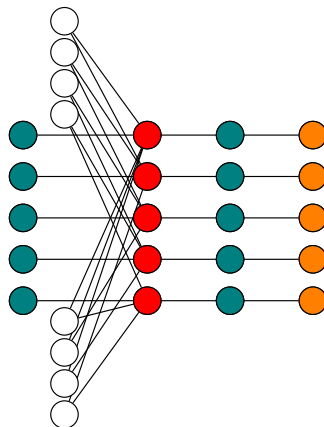
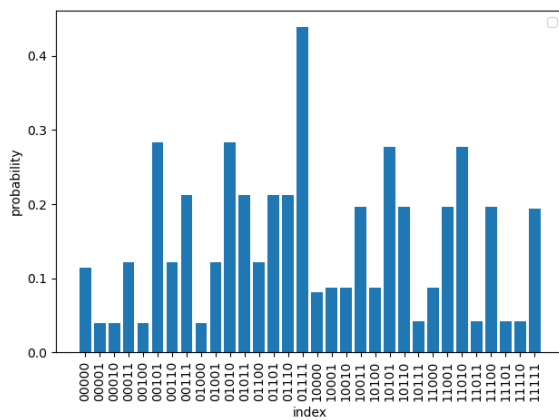


図 3: 頂点被覆問題を解く MB-VQE の回路図の繰り返し部分。上から順に 2 の頂点の番号と対応しており、白い量子ビットは辺の情報をエンコードしている。繰り返し部分一つは CZ ゲート、 R_x 回転、 R_z 回転の 3 つのパーツから成り、各色の頂点でそれぞれ異なる測定をしている。詳細は [3] を参照。

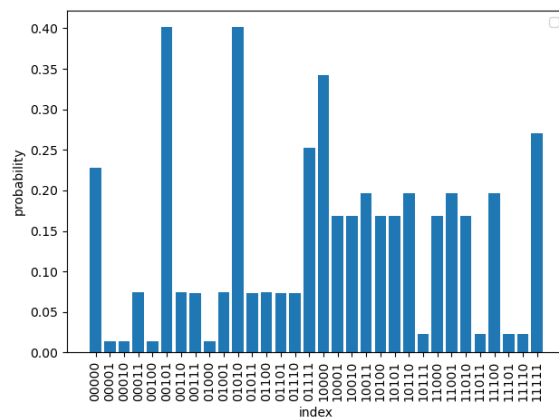
4.2 MB-VQE の手法 (2)

MB-VQE の手法として、ゲート型との差別化のできる、もう一つの手法を紹介します。この手法は、すでに固有値がわかっており、固有状態がグラフ状態を用いて表されるとき、そのハミルトニアンに摂動を与えたときの固有値の変化を調べる時などに有効です。

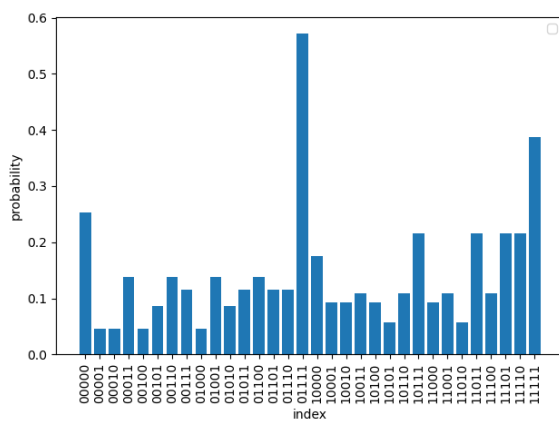
*11 今回行ったシミュレーションは、本物の量子コンピュータを使用したわけではなく、すべて古典コンピュータで実行したものです。量子コンピュータを古典コンピュータでナイーブにシミュレートすると、この実験の例では 32×32 の行列計算が必要となります。これをさらにビットを増やすと、それなりに時間がかかってしまいます。



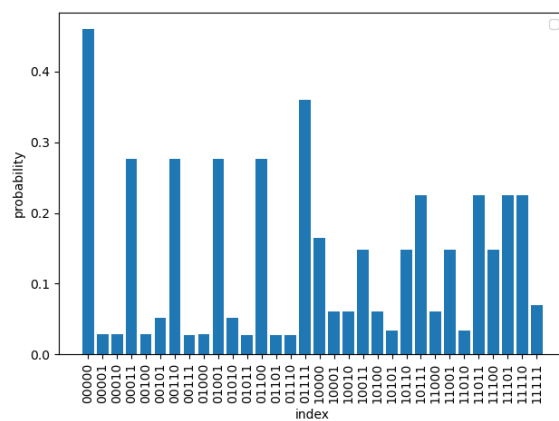
(a) ite=1



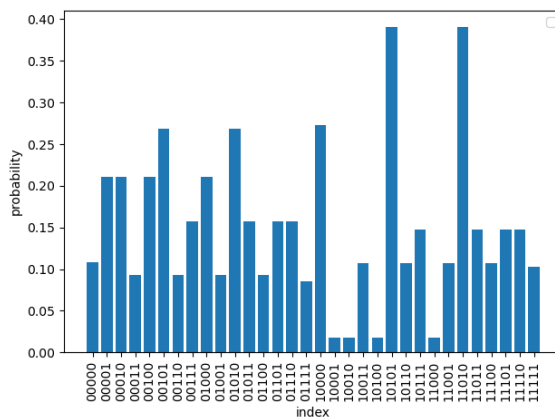
(b) ite=40



(c) ite=80



(d) ite=120



(e) ite=160

図 4: 頂点被覆問題の MB-VQE による解析結果^{*11}。ite は学習のステップ数。回路の繰り返し回数は 8。初めはランダムな状態からスタートして (a)、最終的には解となる状態 $|10101\rangle$, $|11010\rangle$ で最も高い確率をとるような状態が出力される (e)。

例として、トーリック符号ハミルトニアンに一様磁場の摂動を与える問題を考えます。この量子状態やトーリック符号は量子誤り訂正に用いられますが、ここではその詳細は割愛します。数式で記述すると、

$$H_0 = X_1 X_3 X_4 X_5 + X_2 X_4 X_6 X_3 + X_5 X_7 X_1 X_8 + X_6 X_8 X_2 X_7 \quad (47)$$

$$+ Z_7 Z_2 Z_3 Z_1 + Z_8 Z_1 Z_4 Z_2 + Z_3 Z_6 Z_7 Z_5 + Z_4 Z_5 Z_8 Z_6$$

$$H_p = X_1 + X_2 + X_3 + X_4 + X_5 + X_6 + X_7 + X_8 \quad (48)$$

$$H = H_0 + \lambda H_p \quad (49)$$

で表されるハミルトニアンの基底エネルギーを求める問題です。

まず、摂動のないハミルトニアン H_0 の基底状態は、図 5b のようにグラフ状態で表されることが知られています。各点が線で結ばれているところは、CZ ゲートで結ばれている、つまり十分にエンタングルしていることを意味します。一方、摂動ハミルトニアン H_p の基底状態は $|0\rangle^{\otimes 8}$ で表されることは、簡単に確認されます。この状態は一切エンタングルしていません。この事実に注目して、まず図 5b のグラフ状態からスタートし、各辺にエンタングルメントを切り離すような (パラメータで特徴付けられる) 操作を与え、基底エネルギーを求めて最適化していくというアイデアです。

それでは、エンタングルを切り離す操作とは、どのようなものでしょうか。ここでは、基底状態のグラフ状態の各辺を表 5a のように 4 つのパラメータを用いた測定量子ビットを付け足して、各辺ごとにエンタングルの度合いを調整するという方法をとることとします。このように変換することで、24 個の量子ビットを使って最適化を行うことができます。

この方法が 4.1 項で述べた方法と決定的に異なるところは、ゲート型モデルとの対応が自明ではないということです。もちろん任意の測定型回路はゲート型に書き直すことができるため、ゲート型と等価であることに変わりはないですが、4.1 章の場合と異なり基底状態のグラフの特徴に基づいて回路を決定しているため、回路構成の発想が全く異なります。このことが新たな VQE 手法としての可能性を秘めてると考えられます。

5 測定型量子計算のさらなる独自性

ここまでで、測定型量子計算は今まで親しんできたゲート型量子コンピュータと全く異なる発想で計算を進められることを実感できたのではないのでしょうか。ここからは、まだ触れていない測定型のゲート型とのさらなる違いについて、いくつかみてみましょう。

エンタングルの消費が明確である

量子力学のいくつかの不思議な現象には、量子ビットの「エンタングル」が関わっていることが知られています。EMC 系で書かれた測定型量子計算の中では、エンタングルを用意しているのは最初の E のステップだけで、実際に計算を実行している M のステップではエンタングルを壊しているにすぎません。エンタングルを十分に準備して破壊していくことで計算が進むことから、この計算モデルは一方向量子コンピュータ (one-way computer) とも呼

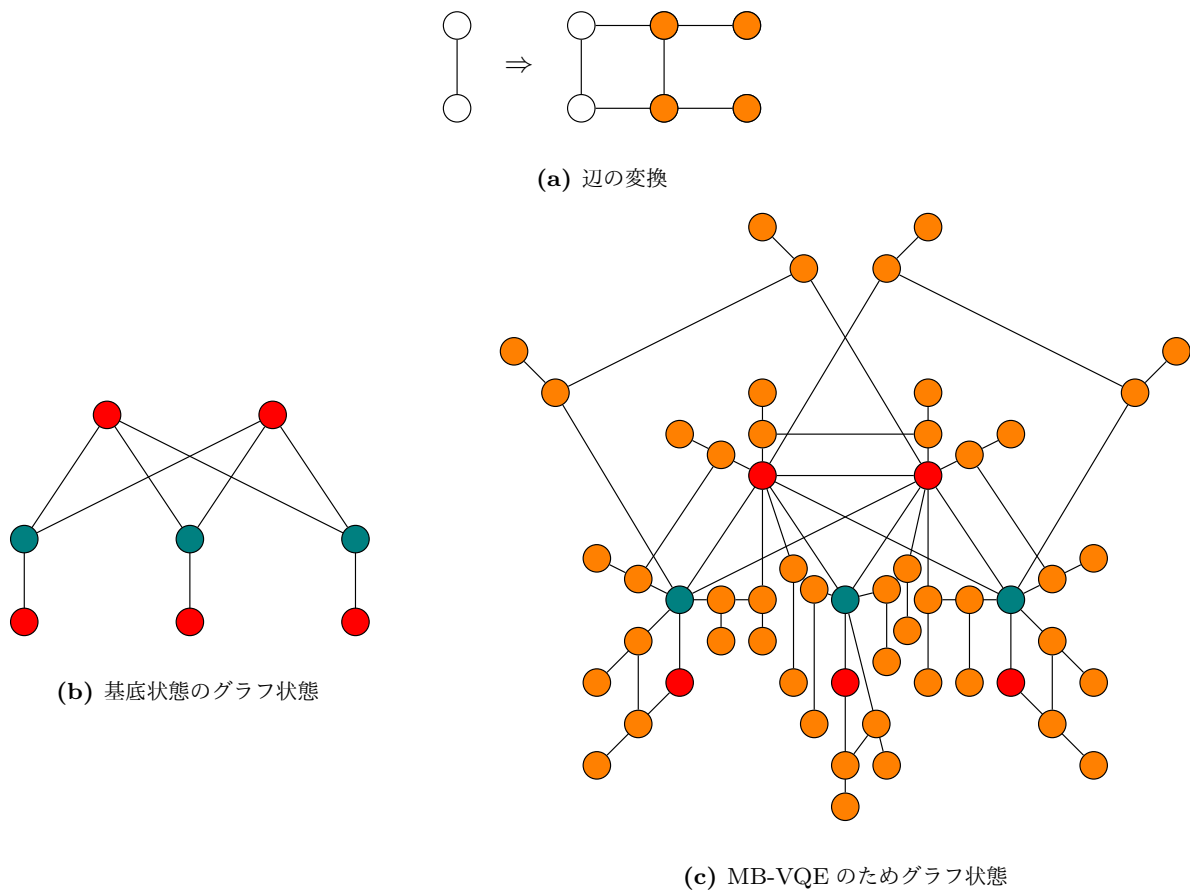


図 5: トーリック符号のハミルトニアンに摂動を与える問題に対する MB-VQE。(a) 摂動前のハミルトニアンの基底状態。(b) 摂動を与えた後の基底状態を見つけるためのグラフ状態。詳細は [5] を参照。

ばれています。それに対し、通常のゲート型では、回路のあらゆる部分にエンタングルを扱う操作が隠れており、エンタングルの度合いが不透明なまま計算が進んでしまいます。

エンタングルに失敗しても良い

ゲート型量子計算で意味のある計算をしようとした場合、一般に多数の量子ゲートを使用します。特に 2 量子ビットに対して演算を行うゲートは現在の技術では成功確率はそれほど高くないですが、もし計算の終盤で失敗してしまったら最初からやり直さなければならず、大変です。それに対して測定型では、最初のリソース状態を準備する段階では、CZ ゲートによるエンタングルを失敗しても、まだ計算 (測定) を行う前の段階ですし、その部分だけを捨ててまた別の量子ビットとエンタングルを試みれば良いのです。

実装の試み

測定型量子計算の実装についても多くの研究が進められています。特に測定型における量子コンピュータの実装に対して必要なことは、(1) 大きな安定したリソース状態 (多数の量子

ビットがエンタングルした状態)を作成すること、(2) 量子ビットの一つ一つに対して射影測定を行うこと、の2点です。これらの点に関して、光ポンピングなどを用いて量子ビットに対応する $^{40}\text{Ca}^+$ の状態を制御するイオントラップ量子コンピュータ [6] や、スクイーズされた光を利用した光測定型量子コンピュータ [7] などが提案されています。

6 最後に

本稿では、近年話題の量子コンピュータの中で、よく知られているゲート型とは異なる測定型の計算方法について、ユニバーサリティや VQE への応用を触れながら基礎を解説しました。今回紹介したいくつかの話題には比較的新しい内容も含まれています。また、本稿では紹介していませんが、測定型量子計算を用いたフォールトトレラント量子計算もホットな話題です。興味を持った方は、参考文献にある記事を含め、ぜひ様々な文献を読んでみることをお勧めします。現在は量子計算といえばゲート型の方が有名ですが、将来測定型が量子計算のスタンダードになるときが来るかもしれません。

参考文献

- [1] <https://qiskit.org/textbook/ja/ch-applications/vqe-molecules.html>.
- [2] <https://amplify.fixstars.com/ja/techresources/research/ising-model-formulation/vertex-covering/>.
- [3] <https://qml.baidu.com/tutorials/measurement-based-quantum-computation/polynomial-unconstrained-boolean-optimization-problem-in-mbqc.html>.
- [4] Vincent Danos, Elham Kashefi, and Prakash Panangaden. The Measurement Calculus. *arXiv:0704.1263 [quant-ph]*, April 2007. arXiv: 0704.1263.
- [5] Ryan R. Ferguson, Luca Dellantonio, Karl Jansen, Abdulrahim Al Balushi, Wolfgang Dür, and Christine A. Muschik. A measurement-based variational quantum eigensolver. *arXiv:2010.13940 [hep-lat, physics:quant-ph]*, March 2021. arXiv: 2010.13940.
- [6] B. P. Lanyon, P. Jurcevic, M. Zwerger, C. Hempel, E. A. Martinez, W. Dür, H. J. Briegel, R. Blatt, and C. F. Roos. Measurement-based quantum computation with trapped ions. *arXiv:1308.5102 [quant-ph]*, August 2013. arXiv: 1308.5102.
- [7] Mikkel V. Larsen, Xueshi Guo, Casper R. Breum, Jonas S. Neergaard-Nielsen, and Ulrik L. Andersen. Deterministic multi-mode gates on a scalable photonic quantum computing platform. *Nature Physics*, 17(9):1018–1023, 2021. Publisher Copyright: © 2021, The Author(s), under exclusive licence to Springer Nature Limited.
- [8] Michael A. Nielsen and Isaac L. Chuang. *Quantum Computation and Quantum Information: 10th Anniversary Edition*. Cambridge University Press, 2010.